Integration Guide
Revision C

McAfee One Time Password

COPYRIGHT
Copyright © 2014 McAfee, Inc. Do not copy without permission.

TRADEMARK ATTRIBUTIONS
McAfee, the McAfee logo, McAfee Active Protection, McAfee DeepSAFE, ePolicy Orchestrator, McAfee ePO, McAfee EMM, Foundscore, Foundstone, Policy Lab, McAfee QuickClean, Safe Eyes, McAfee SECURE, SecureOS, McAfee Shredder, SiteAdvisor, McAfee Stinger, McAfee Total Protection, TrustedSource, VirusScan, WaveSecure are trademarks or registered trademarks of McAfee, Inc. or its subsidiaries in the United States and other countries. Other names and brands may be claimed as the property of others.

Product and feature names and descriptions are subject to change without notice. Please visit mcafee.com for the most current products and features.

LICENSE INFORMATION

License Agreement
NOTICE TO ALL USERS: CAREFULLY READ THE APPROPRIATE LEGAL AGREEMENT CORRESPONDING TO THE LICENSE YOU PURCHASED, WHICH SETS FORTH THE GENERAL TERMS AND CONDITIONS FOR THE USE OF THE LICENSED SOFTWARE. IF YOU DO NOT KNOW WHICH TYPE OF LICENSE YOU HAVE ACQUIRED, PLEASE CONSULT THE SALES AND OTHER RELATED LICENSE GRANT OR PURCHASE ORDER DOCUMENTS THAT ACCOMPANY YOUR SOFTWARE PACKAGING OR THAT YOU HAVE RECEIVED SEPARATELY AS PART OF THE PURCHASE (AS A BOOKLET, A FILE ON THE PRODUCT CD, OR A FILE AVAILABLE ON THE WEBSITE FROM WHICH YOU DOWNLOADED THE SOFTWARE PACKAGE). IF YOU DO NOT AGREE TO ALL OF THE TERMS SET FORTH IN THE AGREEMENT, DO NOT INSTALL THE SOFTWARE. IF APPLICABLE, YOU MAY RETURN THE PRODUCT TO MCAFEE OR THE PLACE OF PURCHASE FOR A FULL REFUND.

Contents

Preface 7
About this guide . . . 7
Conventions . . . 7
Find product documentation . . . 8

1 Introduction to the integration modules 9

2 Citrix XenApp Web Interface integration module 11
Citrix XenApp Web Interface . . . 11
Requirements for integrating with Citrix XenApp Web Interface . . . 11
Install the Citrix XenApp Web Interface integration module . . . 11
Configure Citrix XenApp Web Interface . . . 12
Configure McAfee OTP to integrate with Citrix XenApp Web Interface . . . 13
Test the Citrix XenApp Web Interface integration . . . 14

3 Remote Desktop Web Access 15
Requirements for integrating with RD Web Access . . . 15
Install the RD Web Access integration module . . . 16
Create the RDWebIISIntegration virtual directory . . . 16
Edit the RD Web Access web.config file . . . 16
Configure the RD Web Access filter . . . 17
RD Web Access registry keys . . . 18
Configure McAfee OTP to integrate with RD Web Access . . . 20

4 Microsoft Outlook integration 23
Microsoft Outlook Web App . . . 23
Requirements for integrating with Outlook Web App . . . 23
Install the Outlook Web App integration module . . . 24
Create the Outlook Web App virtual directory . . . 24
Edit the Outlook Web App 2013 web.config file . . . 24
Configure the Outlook Web App 2013 filter . . . 25
Outlook Web App 2013 registry keys . . . 25
Configure Microsoft Exchange for Outlook Web App 2013 integration . . . 27
Configure McAfee OTP for Outlook Web App 2013 integration . . . 28
IIS Secure Access Filter . . . 31
Requirements for integrating with IIS Secure Access Filter . . . 32
Install the IIS Secure Access Filter integration module . . . 32
Configure the filter for IIS Secure Access Filter . . . 35
Set the IIS Secure Access Filter authentication . . . 36
McAfee OTP configuration information with IIS Secure Access Filter . . . 36

5 Microsoft SharePoint integration modules 37
Considerations for using forms authentication with SharePoint . . . 37
Microsoft SharePoint 2010 . . . 37
How McAfee OTP and Microsoft SharePoint work together . . . 38

Requirements for integrating with Microsoft SharePoint 2010 . . . 38
Configure SharePoint 2010 to integrate with McAfee OTP . . . 39
Install the SharePoint 2010 integration module . . . 39
Configure the integration module . . . 40
Test the McAfee OTP and SharePoint 2010 integration . . . 42

6 Microsoft Internet Information Services integration modules 43

Internet Information Service (IIS) 7.x Custom AD Membership Provider . . . 43
Requirements for integrating with IIS Custom Site . . . 43
Install the IIS Custom Site integration module . . . 44
Configuring IIS Custom Site . . . 44
Test the web application . . . 51
Customize the Login page . . . 51
Microsoft ISA Server 2006 . . . 51
Requirements for integrating with Microsoft ISA Server 2006 . . . 51
Install the Microsoft ISA Server 2006 integration module . . . 52
Edit the Windows registry file with McAfee OTP information . . . 52
Configure the ISA Server . . . 52
Configure McAfee OTP to integrate with Microsoft ISA Server 2006 . . . 53
Troubleshooting the ISA Server 2006 integration . . . 55
Microsoft Forefront Threat Management Gateway . . . 55
Requirements for integrating Microsoft Forefront TMG . . . 55
Install the Microsoft Forefront TMG integration files . . . 55
Configure Microsoft Forefront TMG parameters to integrate with McAfee OTP . . . 56
Configure Microsoft Forefront TMG for integration . . . 56
Configure Microsoft Forefront TMG to show McAfee OTP authentication filter . . . 57
Configure McAfee OTP for Microsoft Forefront TMG . . . 57

7 Miscellaneous integration modules 61

Lotus Domino 6 . . . 61
Requirements for integrating with Domino . . . 61
Install the Domino installation package . . . 62
Configure McAfee OTP for LDAP lookup . . . 63

8 Overview of the McAfee OTP APIs 65

An introduction to the Java APIs available . . . 66
Set up a Java development environment example . . . 66
The McAfee OTP plug-in API . . . 67
The DBHandler class . . . 68
Using the OTPAddRADIUSAttribute API . . . 68
The OTPLogging API . . . 69
The OTPVerificationHandler API . . . 70
The McAfee OTP Native Client API . . . 70
McAfee OTP API error codes . . . 70
The McAfee OTP Microsoft .NET client plug-in . . . 71
Requirements for integrating with the .NET client plug-in . . . 71
Install the .NET client plug-in . . . 72
Use the .NET client with Visual Studio .NET . . . 72
McAfee OTP API methods . . . 72
McAfee OTP .NET client API error codes . . . 73
Web Services SOAP API . . . 74
Set up the Web Services SOAP API . . . 74
Information to integrate McAfee OTP with the Web Services SOAP API . . . 74
SOAP operations . . . 75
Commands . . . 78
Client code examples . . . 88

Index **91**

Contents

Preface

This guide provides the information you need to work with your McAfee product.

Contents

- *About this guide*
- *Find product documentation*

About this guide

This information describes the guide's target audience, the typographical conventions and icons used in this guide, and how the guide is organized.

Conventions

This guide uses these typographical conventions and icons.

Book title, *term*, *emphasis*	Title of a book, chapter, or topic; a new term; emphasis.
Bold	Text that is strongly emphasized.
`User input, code, message`	Commands and other text that the user types; a code sample; a displayed message.
Interface text	Words from the product interface like options, menus, buttons, and dialog boxes.
Hypertext blue	A link to a topic or to an external website.
	Note: Additional information, like an alternate method of accessing an option.
	Tip: Suggestions and recommendations.
	Important/Caution: Valuable advice to protect your computer system, software installation, network, business, or data.
	Warning: Critical advice to prevent bodily harm when using a hardware product.

Find product documentation

After a product is released, information about the product is entered into the McAfee online Knowledge Center.

Task

1 Go to the McAfee ServicePortal at http://support.mcafee.com and click **Knowledge Center**.

2 Enter a product name, select a version, then click **Search** to display a list of documents.

1 Introduction to the integration modules

You can integrate McAfee® One Time Password (McAfee OTP) with applications and systems through integration modules and protocols.

McAfee OTP can be integrated with most VPN services using the RADIUS protocol. Because McAfee OTP can act as a RADIUS server, most VPN/RADIUS-aware products can be integrated without any installation. Configuring the McAfee OTP and the VPN/RADIUS product completes the integration.

Using Java, COM, .NET, and PHP client APIs, you can write custom integration modules for your applications. By using the client APIs, you can add strong authentication to your custom applications.

2 Citrix XenApp Web Interface integration module

Integrate Citrix XenApp Web Interface 5.4 with McAfee OTP to enable strong password authentication for your Citrix XenApp product.

Citrix XenApp Web Interface

To have Citrix XenApp Web Interface protected by McAfee OTP two-factor authentication, download and install the integration module files, and configure Citrix XenApp Web Interface and McAfee OTP so that they can work together.

Requirements for integrating with Citrix XenApp Web Interface

McAfee OTP must be installed and configured before it can integrate with Citrix XenApp Web Interface.

Table 2-1 Minimum requirements

Platform	Version
Citrix	Citrix XenApp Web Interface 5.4
McAfee One Time Password	McAfee OTP 3.5.1 or later

Install the Citrix XenApp Web Interface integration module

Download and install the integration module on your operating system.

Before you begin

The McAfee OTP server must be configured and running correctly before the integration module can be installed.

Task

1 Back up the C:\Inetpub\wwwroot\Citrix web root.

2 Download the installation .zip file from mcafee.nordicedge.com/integrations and save it to C:\Inetpub\wwwroot\Citrix.

If McAfee OTP is installed on a different machine than the Citrix XenApp Web Interface, modify /Xen/auth/loginOTP.aspx to point to the correct server IP address or host name. The default IP address is 127.0.0.1:3100.

3 Locate the web.config file (for example, C:\Inetpub\wwwroot\Citrix\XenApp\web.config) and open it using a text editor.

   a Add the /auth/loginOTP.aspx string to the AUTH:UNPROTECTED_PAGES key.

```
<add key="AUTH:UNPROTECTED_PAGES"
value="/auth/loginOTP.aspx,/rade.asx,...
```

   loginOTP.aspx is case sensitive.

   b In the <appsettings> section, add a value, such as the Citrix server name, to <add key="RADIUS_NAS_IDENTIFIER" value="servername" />.

   The RADIUS_NAS_IDENTIFIER value must be present in the site's web configuration file and can be any string of three characters or more. For example, you can replace "servername" with the name of your server (at least three characters), or keep the string "servername".

Configure Citrix XenApp Web Interface

Configure the Citrix XenApp Web Interface to run on your operating system.

Task

1 Click **Start** | **Programs** | **Citrix** | **Management Consoles** | **Citrix Web Interface Management**.

2 Browse to **XenApp Web Sites** and select the site name where the integration files are installed.

3 From the **XenApp - Edit Settings** menu, click **Authentication Methods**.

The **Configure Authentication Methods** window appears.

4 Select the **Explicit** checkbox, then click **Properties**.

The **Properties** window appears.

5 Select the **Two-Factor Authentication** checkbox.

6 From the **Two-factor setting** drop-down list, select **RADIUS**, then click **Add**.

The **Edit RADIUS Server** window appears.

   a In the **Server address** field, type the McAfee OTP server IP address.

   If McAfee OTP is installed on the same machine, type 127.0.0.1

   b In the **Server port** field, type the McAfee OTP port.

   The default is 1812.

   c Click **OK**.

7 Verify the radius_secret.txt file is located in the C:\Inetpub\wwwroot\Citrix\XenApp\conf folder.

If not, rename the file radius_secret.txt.sample to radius_secret.txt, then edit the file and type the shared secret to be entered in McAfee OTP.

8 Open a command prompt and type `iisreset` to restart IIS.

Configure McAfee OTP to integrate with Citrix XenApp Web Interface

Configure McAfee OTP to use a variety of authentication methods for integration with Citrix XenApp Web Interface.

Task

1 Open the McAfee OTP configuration console.

2 From the context menu or configuration pane, right-click the **Databases** object type, then select the **New LDAP** database type.

3 On the configuration pane, configure the settings.

Option	Definition
Host Settings	• **Database Display Name** — Specifies a unique name. • **Host Address** — Specifies the IP address or DNS name of the LDAP server. • **Port number** — Specifies the port number of the LDAP server. • **Admin DN** — Specifies the DN of an administrative user that has read and write access to the Account Disable attribute for all user accounts. • **Password** — Specifies the password of an administrative user that has read and write access to the Account Disable attribute for all user accounts. • **Test Connection** — Verifies the connection to the LDAP server.
Search Settings	Use the configuration pane to configure these settings: • **Base DN** — Specifies the location in the directory tree from which McAfee OTP searches for users. • **Scope** — Specifies the scope of the directory search. • **No of Connections** — Specifies the maximum number of connections that McAfee OTP can have to the LDAP server. • **Filter Start** — Specifies the beginning of the search filter. • **Filter End** — Specifies the end of the search filter.
Account Settings	**OTP Attribute** — Specifies the LDAP attribute that McAfee OTP uses to look up an email address, or mobile phone number.

4 From the context menu or configuration pane, right-click **Clients** and select the option to create a new RADIUS client.

5 On the configuration pane, configure the settings.

Option	Definition
Name & Address	Use the configuration pane to configure these settings: • **Client Display name** — Specifies a unique name. • **Client IP Address** — Specifies the IP address of the Citrix XenApp Web interface server.
RADIUS Options	Deselect **Supports RADIUS Access-Challenge**, then configure these settings: • **Shared Secret** — Specifies the RADIUS client's shared secret. This is the same as radius_secret.txt • **Auth. Server IP Address** — Specifies the IP address of the Citrix XenApp Web interface server.

6 On the select pane, select **RADIUS**.

a Select the **Enable RADIUS** checkbox.

b Verify the port number is 1812.

7 Click **Save Config**.

The **Save Changes** dialog box appears.

8 Click **Save**.

9 Open a command prompt and type `iisreset` to restart IIS before you use the Citrix XenApp Web Interface with McAfee OTP.

Test the Citrix XenApp Web Interface integration

Test the Citrix XenApp Web Interface integration with McAfee OTP to confirm that the user receives a one-time password that the McAfee OTP can verify it and let the user log on.

Before you begin

Restart the Internet Information Service (IIS).

Task

1 Go to the Citrix XenApp Web Interface.

2 Type your user name, password, and domain, then click **Log On**.

3 Type your one-time password, then click **Log On**.

The application icons appear and the logon is authenticated.

3 Remote Desktop Web Access

The Microsoft Remote Desktop Web Access (RD Web Access) integration module for McAfee One Time Password enables strong authentication. An HTTP module filter protects RD Web Access and communicates with the McAfee OTP server.

Module features:

- Supports RD Web Access Forms authentication
- Installed as an HTTP Module filter to protect all incoming requests
- Includes debug logging (Event Viewer)
- Delivers one-time passwords by:
  - Text message (McAfee SMS)
  - McAfee Pledge software tokens (OATH)
  - Intel IPT (TOTP)
  - Hardware tokens (OATH)
  - SMPP
  - SMTP

Contents

- *Requirements for integrating with RD Web Access*
- *Install the RD Web Access integration module*
- *Create the RDWebIISIntegration virtual directory*
- *Edit the RD Web Access web.config file*
- *Configure the RD Web Access filter*
- *Configure McAfee OTP to integrate with RD Web Access*

Requirements for integrating with RD Web Access

Make sure your system meets these requirements to guarantee a successful integration.

Platform	Version
Microsoft Windows Server	• Microsoft Windows Server 2008 R2 or later • Microsoft .NET Framework 3.5 or later has to be installed on the server
McAfee One Time Password	Version 3.5.1 or later

McAfee OTP can use an LDAP v3-compatible directory service and/or an ODBC-compliant database server to perform authentication and mobile number lookup. Active Directory is the recommended directory service for RD Web Access. Active Directory must be configured for McAfee OTP to authenticate and retrieve mobile numbers for users.

Install the RD Web Access integration module

Download the installation files and add the files to the installation directory.

> **Before you begin**
> Ensure the McAfee OTP server is available. It does not have to be installed on the same machine as the RD Web Access.

Task

1. Download the latest RDWeb.exe integration module.
2. Run OTP_Integration_RDWeb.exe.
3. Unzip the files.

   The default installation path is C:\Program Files\McAfee\OTP_Integrations\RDWeb

   If you change the default installation path, change any references to it (such as in the registry settings).
4. Using administrator rights, double-click OTP_RDWeb_RegistrySettings.reg to create a SOFTWARE \McAfee\One Time Password\RDWeb registry key and its subkeys.
5. Open C:\Windows\Web\RDWeb\Pages\Bin.
6. Copy these files from the installation zip file into the \Bin folder.
   - McAfee.OTP.IIS.dll
   - NordicEdgeOTP.dll

See also

Create the RDWebIISIntegration virtual directory on page 16

Create the RDWebIISIntegration virtual directory

Add the RDWebIISIntegration virtual directory to the IIS default website.

Task

1. In IIS Manager, right-click **Default Web Site** and select **Add Virtual Directory**.
2. Set the alias to `RDWebIISIntegration`.
3. Set the physical path to C:\Program Files\McAfee\OTP_Integrations\RDWeb\RDWebIISIntegration \UI, and click **OK**.

Edit the RD Web Access web.config file

Edit the web.config file and restart the IIS Manager.

Task

1. Go to the folder C:\Windows\Web\RDWeb\Pages
2. Make a copy of the web.config file.
3. Open web.config with a text editor.

4 Locate the tag <modules> and add the following row as the first module in the list:

<add name="HttpAuthenticationModule" type="McAfee.OTP.IIS.HttpAuthenticationModule" />

After editing, the modules section should look like this:

```
<modules runAllManagedModulesForAllRequests="true">
    <remove name="FormsAuthentication" />
    <add name="HttpAuthenticationModule" type="McAfee.OTP.IIS.HttpAuthenticationModule" />
    <add name="RDWAFormsAuthenticationModule"
type="Microsoft.TerminalServices.Publishing.Portal.FormAuthentication.TSFormsAuthenticatio
n" />
</modules>
```

5 Make sure that forms-based authentication is configured:

```
<authentication mode="Forms">
   <forms loginUrl="default.aspx" name="TSWAAuthHttpOnlyCookie" protection="All"
requireSSL="true" />
</authentication>
```

6 Save the web.config file.

7 Using Administrator permissions, open a command prompt and type `iisreset` to restart IIS.

See also

RD Web Access registry keys on page 18

Configure the RD Web Access filter

All settings for the filter are defined in the Windows registry. If any keys are missing, default values will be used by the filter.

Several keys specify that URL or file paths must be valid for the filter to run properly. All file paths used by the filter must have the necessary access rights.

Task

1 On your McAfee OTP server, click **Start** | **Run**.

2 Type `regedit` and click **OK**.

3 Browse to HKLM SOFTWARE\McAfee\One Time Password.

Most of the predefined key values do not have to be modified; however, some values are specific for your environment, like StaticLogonDomain (this is simply your AD domain) and the OtpServerAddress.

4 Configure SmsClientDetectionName, PledgeClientDetectionName and/or EmailClientDetectionName according to your McAfee OTP settings.

The registry configuration is read at the web application startup. If you change the configuration, you must restart the web application

RD Web Access registry keys

HKLM SOFTWARE\McAfee\One Time Password

Table 3-1 HKLM SOFTWARE\McAfee\One Time Password.

Key	Default Value	Description
SessionManagerDebug	0	If set to 1, a log is in **Event Viewer** | **Windows Logs** | **Application**. Look for **SessionManager** in the **Source** column.

Registry keys for HKLM SOFTWARE\McAfee\One Time Password\RDWeb.

Table 3-2 HKLM SOFTWARE\McAfee\One Time Password\RDWeb

Key	Default Value	Description
ChangeADPasswordURL	http:// ChangeADPasswordURL	If McAfee OTP detects that a user password is about to expire, the user is redirected to the URL given in this key.
CredentialsPostURL	/RDWeb/Pages/en-US/ login.aspx	A URL to which user credentials are posted after a successful two-factor authentication.
EmailClientDetectionName	[empty]	Example: EMAIL
Encryption	1	DES encryption between the client and the server. • 0 = No encryption. • 1 = Encryption.
EventViewerDebug	0	Opens a debug log. • 0 = A debug log is not created. • 1 = A debug log is created. The debug log is in **Event Viewer** | **Windows Logs** | **Application**. In the **Source** column, look for **HttpAuthenticationModule**. The key SessionManagerDebug must be set to 1 to use Session Manager debugging. If no log entries are written to the event viewer, try to create the **Source** name manually using the command prompt: C:\Windows \system32>eventcreate /ID 1 /L APPLICATION /T INFORMATION /SO HttpAuthenticationModule /D "My first log"
ExcludedPages	login.aspx expiredpassword.aspx	Pages in RD Web Access that are excluded from the filter.
IgnoredURLs	.css .xml .jpg	If the given string is included in the URL, it is ignored by the filter.
MaxSessions	10000	The maximum number of sessions that can exist in the module session store.

Table 3-2 HKLM SOFTWARE\McAfee\One Time Password\RDWeb *(continued)*

Key	Default Value	Description
KeepSessions	9000	Specifies the number of the current sessions that will be kept after MaxSessions has been reached.
OtpIntegrationFilePath	`C:\Program Files \McAfee\OTP _Integrations\RDWeb \RDWebIISIntegration \`	The path to the directory containing the McAfee OTP integration files and directories.
OtpIntegrationIISWebAppName	RDWebIISIntegration	The name of the web application (Virtual Directory) where HTML components for the integration module are located such as images, HTML pages, style sheets, and java scripts.
OtpServerAddress	127.0.0.1:3100	Sets the address of the McAfee OTP server. Either a plain host name or multiple host names/port numbers for failover with the following syntax: 192.168.10.3:3100;otp.acme.com: 3567;otpserver.xyz.com:3100 Use colons (:) to separate host name from port number and semicolon (;) to separate multiple McAfee OTP servers.
PledgeClientDetectionName	[empty]	Example: PLEDGE
PostURL	`/RDWeb/Pages/`	A URL where UPLogin.html and OTPLogin.html are posted.
RDPCertificates	Example: B8A4D3D56C6B271E44B25F17A29CB7D7D6975BD5	
	[empty]	To configure this key, use the instructions in this Microsoft link, or follow these steps: 1 Before you enable the integration module, open Internet Explorer from a web client. 2 Go to https://[server.domain]/RDWeb/Pages/en-US/login.aspx. 3 Right-click the RD Web Access logon page, and select **View source**. 4 Find the <input type ="hidden" name ="RDPCertificates"> string. 5 Get the following value, and set it as the value of RDPCertificates.
	Value: `value=B8A4D3D56C6B271E44B25F17A29CB7D7D6975BD5`	
RemoveOldSessionsInterval	5	Value in minutes. Removes sessions that are not used (sessionsToRemove = MaxSessions - KeepSessions).

Table 3-2 HKLM SOFTWARE\McAfee\One Time Password\RDWeb *(continued)*

Key	Default Value	Description
RemovePrivatePublicButtons	0	If set to 1, the Private Computer and Public Computer timeout options are removed from the logon form.
SessionTimeOut	0	Integration module session timeout in minutes. RD Web Access has its own session timeouts. These values can be changed in `C:\Windows\Web\RDWeb\Pages\web.config`. On the RD Web Access logon page, these are shown as two radio buttons. • A public or shared computer that times out after 20 minutes • A private computer that times out after 4 hours See http://technet.microsoft.com/en-us/library/cc731508.aspx for information about session timeouts.
SmsClientDetectionName	[empty]	Example: SMS
StaticLogonDomain	[empty]	Example: MyADDomain

Configure McAfee OTP to integrate with RD Web Access

Configure the RD Web Access integration module to use a variety of authentication methods for integration with RD Web Access. This example integrates RD Web Access for text message and Pledge authentication.

You can configure more than one authentication method but you must configure *at least* one method.

Task

1 In the McAfee OTP configuration console, create a new database with a unique name for Pledge authentication.

2 Configure the database with these settings.

Table 3-3 Option definitions - Host Settings

Option	Definition
Host Address	The IP address or DNS name of the LDAP server.
Port number	The port number of the LDAP server.
Admin DN	The DN of an administrative user that has read and write access to the Account Disable attribute for all user accounts.

Table 3-4 Option definitions - Search Settings

Option	Definition
Base DN	Specifies the directory tree location that McAfee OTP uses to search for users.
Scope	The scope of the directory search.
No. of connections	The maximum number of connections that McAfee OTP can have to the LDAP server.

Table 3-4 Option definitions - Search Settings *(continued)*

Option	Definition
Filter Start	The beginning of the search filter.
Filter End	The end of the search filter.

a Set the OATH Key to be a multi-value string attribute.

b Test the connection to the database and that LDAP is authenticating correctly.

c Click **Save Config**.

3 Create a new McAfee OTP database with a unique name for text message authentication.

4 Configure the database with these settings.

Table 3-5 Option definitions — Host Settings

Option	Definition
Host Address	The IP address or DNS name of the LDAP server.
Port number	The port number of the LDAP server.
Admin DN	The DN of an administrative user that has read and write access to the Account Disable attribute for all user accounts.

Table 3-6 Option definitions — Search Settings

Option	Definition
Base DN	The location in the directory tree from which McAfee OTP searches for users.
Scope	The scope of the directory search.
No of connections	The maximum number of connections that McAfee OTP can have to the LDAP server.
Filter Start	The beginning of the search filter.
Filter End	The end of the search filter.

Table 3-7 Option definitions — Account Settings

Option	Definition
OTP Attribute	You can choose a single value string attribute.

a Test the connection to the database and that LDAP is authenticating correctly.

b Click **Save Config**.

5 Create a new McAfee OTP server client with a unique name for Pledge authentication with these settings.

Table 3-8 Option definitions — Pledge client settings

Option	Definition
Client IP Address	The IP address of the client.
User Database	Select the database you created for Pledge support.
Client name detection	Type `Pledge`.

6 Create a new McAfee OTP server client with a unique name for text message authentication with these settings.

Table 3-9 Option definitions — Text message client settings

Option	Definition
Client IP Address	The IP address of the client.
User Database	Select the database you created for text message support.
Client name detection	Type `SMS`.

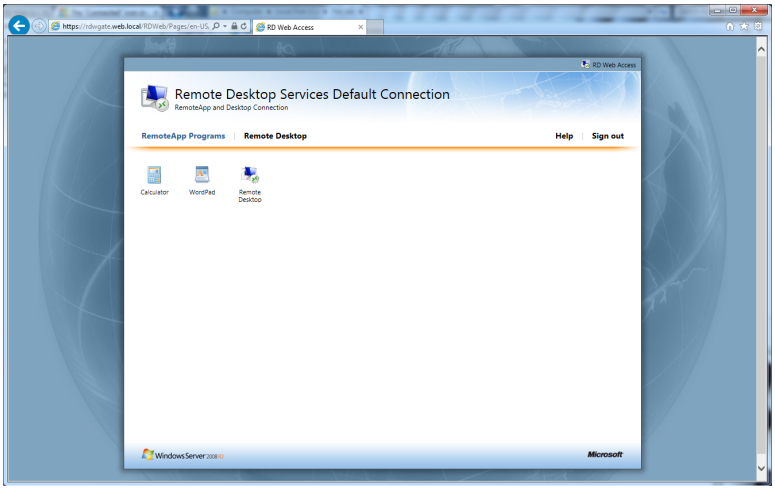
a Set the registry values in the Filter Configuration section according to your McAfee OTP client name settings for:

- SmsClientDetectionName
- PledgeClientDetectionName
- EmailClientDetectionName

b Click **Save Config**.

7 Restart RD Web Access and log on to McAfee OTP to confirm the configuration works.

- If the integration was successful, RD Web Access looks like this.



- If the integration was unsuccessful, the **Connected** icon does not appear in the notification area when you are prompted for your user name and password. To resolve this issue, modify the C:\windows \Web\RDWeb\Pages\renderscripts.js script file using information in http://support.microsoft.com/default.aspx?scid=kb;EN-US;977507.

4 Microsoft Outlook integration

The Microsoft Outlook Web App 2013 integration module for McAfee OTP enables strong two-factor authentication for Microsoft Outlook. An ISAPI filter and IIS Secure Access Filter, protects the Outlook Web App and communicates with McAfee OTP.

Contents

- *Microsoft Outlook Web App*
- *IIS Secure Access Filter*

Microsoft Outlook Web App

An HTTP module filter protects the Outlook Web App application and communicates with the McAfee OTP server.

The Outlook Web App integration module is installed as an HTTP Module filter to protect all incoming requests and:

- Supports Outlook Web App Forms authentication
- Performs debug logging through the event viewer
- Delivers one-time passwords by:
  - Text message (McAfee SMS)
  - McAfee Pledge software tokens (OATH)
  - Intel IPT (TOTP)
  - Hardware tokens (OATH)
  - SMPP
  - SMTP

Requirements for integrating with Outlook Web App

Make sure your system meets these requirements to guarantee a successful integration.

Table 4-1 Minimum requirements

Platform	Version
Microsoft Windows Server	Microsoft Windows Server 2003 or later Microsoft .NET Framework 3.5 or later has to be installed on the Microsoft Windows Server.
Microsoft Exchange	Microsoft Exchange Server 2013
McAfee One Time Password	Version 3.5.1 or later

Active Directory has to be configured for McAfee OTP to authenticate and retrieve mobile numbers for users.

McAfee OTP can use any LDAP v3-compatible directory service and/or an ODBC-compliant database server to perform authentication and mobile number lookup. Active Directory is the recommended directory service for Microsoft Outlook Web App.

Install the Outlook Web App integration module

Download the installation files and add the files to the installation directory.

Before you begin

The Microsoft Exchange server is running correctly and that the McAfee OTP server is available.

McAfee OTP does not have to be installed on the same server as Microsoft Outlook.

Task

1. Download the latest integration module for the Outlook Web App integration module that you are using.
2. Run OTP_Integration_OWA_2013.exe and unzip the files to the Outlook Web App path in Program Files.

   The default installation path is C:\Progam Files\McAfee\OTP_Integrations\OWA
3. As an administrator, double-click OTP_OWA2013_RegistrySettings.reg to create the SOFTWARE \McAfee\One Time Password\OWA and its subkeys.
4. Open C:\Program Files\Microsoft Exchange Server\2013\FrontEnd\HttpProxy\owa\bin.
5. Copy these files from the installation zip file into the \bin folder:
   - McAfee.OTP.IIS.dll
   - NordicEdgeOTP.dll

Create the Outlook Web App virtual directory

Integrate Outlook Web App into IIS.

Task

1. In IIS Manager, right-click the default web site and click **Add Virtual Directory**.
2. Set **Alias** to **OWAIISIntegration**.
3. Set **Physical path** to C:\Program Files\McAfee\OTP_Integrations\OWA\OWAIISIntegration\UI
4. Click **OK** to save the changes.

Edit the Outlook Web App 2013 web.config file

Task

1. Go to the folder C:\Program Files\Microsoft\ExchangeServer\2013\FrontEnd\HttpProxy\owa.
2. Make a copy of the web.config file.
3. Open the web.config file with a text editor.

4 Locate the tag <modules> and add the following row as the first module in the list.

<add name="HttpAuthenticationModule" type="McAfee.OTP.IIS.HttpAuthenticationModule" />

5 Save the web.config file.

After editing the modules section, should look like this:

```
<modules>
   <remove name="ServiceModel" />
   <remove name="ServiceModel-4.0" />
   <remove name="Session" />
   <remove name="Profile" />
   <add name="HttpAuthenticationModule" type="McAfee.OTP.IIS.HttpAuthenticationModule" />
   <add name="HttpProxy"
type="Microsoft.Exchange.HttpProxy.FbaModule,Microsoft.Exchange.FrontEndHttpProxy,Version=201
3,Culture=neutral,PublicKeyToken=31bf3856ad364e35" preCondition="" />
   <add name="cafe_exppw" />
</modules>
```

Configure the Outlook Web App 2013 filter

All settings for the filter are defined in the Windows registry. If any keys are missing, the filter uses default values.

Before you begin

Ensure that URL or file paths are valid and that all paths have the necessary access rights.

Most of the predefined key values do not have to be modified but there are some values that are specific for your environment such as **StaticLogonDomain** which is your Active Directory domain and the **OtpServerAddress**. Remember to configure **SmsClientDetectionName**, **PledgeClientDetectionName** and **EmailClientDetectionName** according to your McAfee OTP server settings.

The registry configuration is read at the web application startup which means that the web application requires to be restarted if the configuration is changed.

Task

1 On the server where Outlook Web App is running, click **Start** | **Run**.

2 Type `regedit`, and click **OK**.

3 Go to HKLMSOFTWARE\McAfee\One Time Password.

4 Edit the registry keys.

5 Restart the web application.

Outlook Web App 2013 registry keys

Table 4-2 HKLM SOFTWARE\McAfee\One Time Password

Key	Default Value	Description
SessionManagerDebug	0	If set to 1, a log is in the **Event Viewer** | **Windows Logs** | **Application**. Look for **SessionManager** in the **Source** column.

Table 4-3 HKLM SOFTWARE\McAfee\One Time Password\OWA

Key	Default Value	Description
ChangeADPasswordURL	http:// ChangeADPasswordURL	If McAfee OTP detects that a user password is about to expire, the user is redirected to the URL that is configured in this key.
CredentialsPostURL	/owa/auth/owaauth.dll	A URL where user credentials are posted after a successful two-factor authentication.
EmailClientDetectionName	[empty]	Example: EMAIL
Encryption	1	DES encryption between the client and the server. • 0 = No encryption. • 1 = Encryption.
EventViewerDebug	0	Opens a debug log. • 0 = A debug log is not created. • 1 = A debug log is created. The debug log is in **Event Viewer** | **Windows Logs** | **Application**. In the **Source** column, look for **HttpAuthenticationModule**. The SessionManagerDebug key also has to be set to 1 if Session Manager debugging is needed. If no log entries are written to the event viewer, try to create the **Source** name manually using the command prompt: C:\Windows \system32>eventcreate /ID 1 /L APPLICATION /T INFORMATION /SO HttpAut henticationModule /D "My first log"
ExcludedPages	logon.aspx expiredpassword.aspx	Pages in the Outlook Web App that are excluded from the filter.
IgnoredURLs	owa/service.svc?action	If the given string is included in the URL, it is ignored by the filter.
MaxSessions	10000	The maximum number of sessions that can exist in the module session store.
KeepSessions	9000	Specifies the number of the current sessions that will be kept after MaxSessions has been reached.
OtpIntegrationFilePath	C:\Program Files\McAfee \ OTP_Integrations\OWA \OWAIISIntegration\	The path to the directory containing the McAfee OTP integration files and directories.
OtpIntegrationIISWebAppName	OWAIISIntegration	The name of the web application (Virtual Directory) where HTML components for the integration module are located (such as images, HTML pages, style sheets, and java scripts).

Table 4-3 HKLM SOFTWARE\McAfee\One Time Password\OWA *(continued)*

Key	Default Value	Description
OtpServerAddress	127.0.0.1:3100	Sets the address of the McAfee OTP server. Either a plain host name or multiple host names/port numbers for failover with the following syntax: 192.168.10.3:3100;otp.acme.com:3567;otpserver.xyz.com:3100 Use colons (:) to separate host name from port number and semicolon (;) to separate multiple McAfee OTP servers.
PledgeClientDetectionName	[empty]	Example: PLEDGE
PostURL	/owa/auth/owaauth.dll	A URL where UPLogin.html and OTPLogin.html are posted.
RemoveOldSessionsInterval	5	Value in minutes. Removes sessions that are not used (sessionsToRemove = MaxSessions - KeepSessions).
RemovePrivatePublicButtons	0	If set to 1, the **Private Computer** and **Public Computer** timeout options are removed from the logon form.
SessionTimeOut	5	On the Outlook Web App logon page, these are shown as two radio buttons with default options. • A public or shared computer that times out after 20 minutes. • A private computer that times out after 4 hours. See http://technet.microsoft.com/en-us/library/cc731508.aspx for information about session timeouts.
SmsClientDetectionName	[empty]	Example: SMS
StaticLogonDomain	[empty]	Example: MyADDomain

Configure Microsoft Exchange for Outlook Web App 2013 integration

Configure Microsoft Exchange so that the Outlook Web App 2013 integration module can use forms-based authentication.

Experimenting with permissions and settings for Microsoft Exchange can seriously damage your installation. If the filter does not work as expected, test without the filter to verify that Microsoft Exchange is working as expected.

Task

1 In Microsoft Exchange, browse to https://<server>/ecp and log on.

2 Click **servers** | **virtual directories**

3 Select your Outlook Web App, and type `OWA` in **Select type**.

4 In the Outlook Web App, ensure that **Use forms-based authentication** is selected.

5 Set the **Logon format** to **Domain/user name**.

6 Click **Save**.

7 Open a command prompt as an administrator and type `iisreset` to restart the IIS web server.

Configure McAfee OTP for Outlook Web App 2013 integration

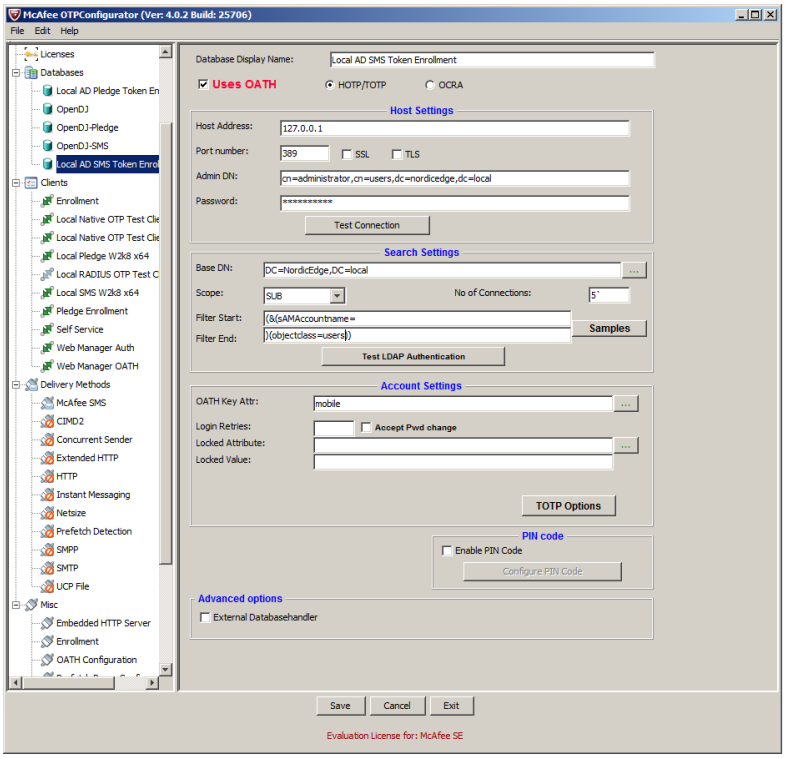
Configure McAfee OTP to use a variety of authentication methods for integration with Outlook Web App 2013. This example integrates Outlook Web App 2013 for text message and Pledge authentication.

You must set up at least one authentication method for the Outlook Web App integration to work.

Task

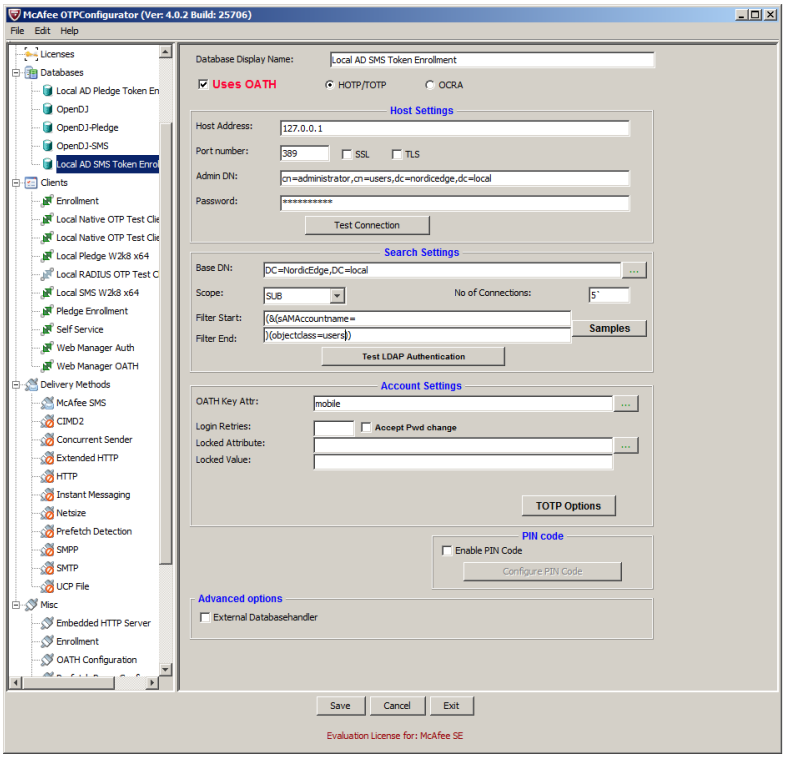
1 In McAfee OTP, open the configuration console.

2 In **Databases**, create a new database to allow authentication with McAfee Pledge. Copy the settings.



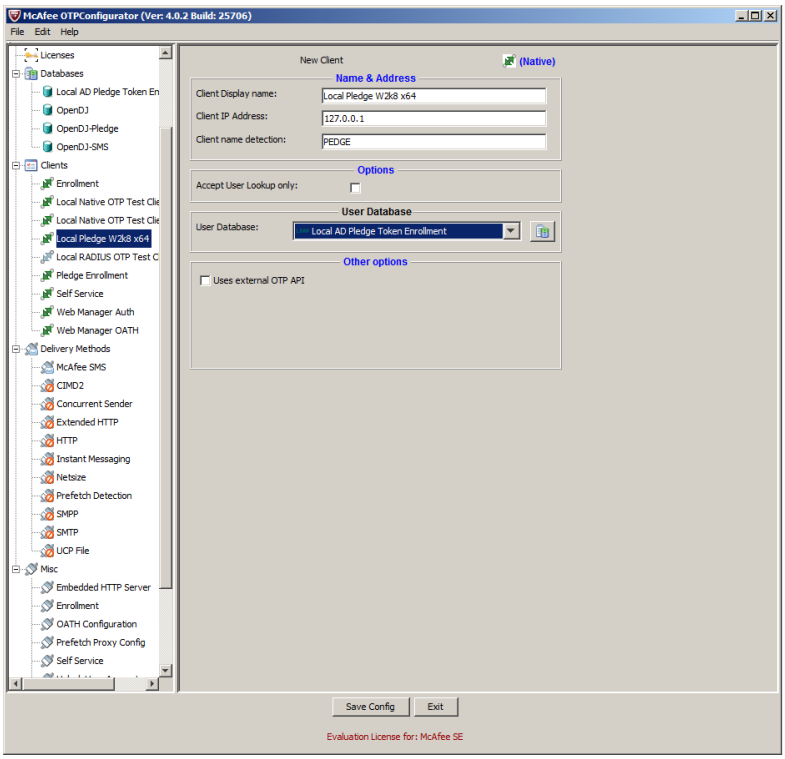
The OATH Key in the **Account Settings** section must be a multiple value string attribute.

3 In **Databases**, create a new database to allow authentication with text message. Copy the settings.



The OTP Attribute in the **Account Settings** section can be a single value string attribute.

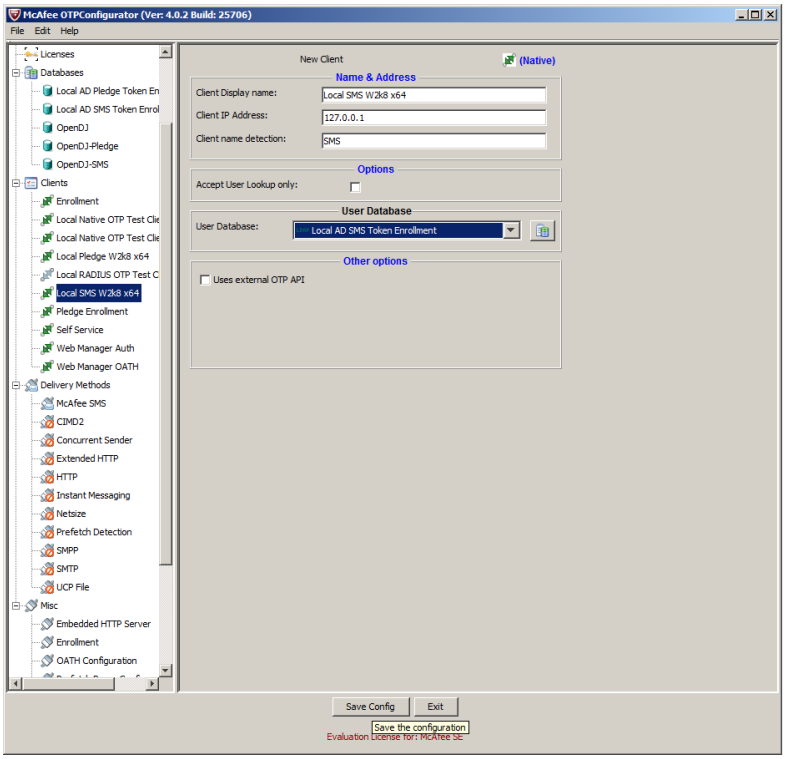
4 In **Clients**, create a new McAfee OTP client to allow authentication with McAfee Pledge. Copy the settings.



5 In **Client name detection**, type a unique name for the new client.

6 In **User Database**, select the database you created for Pledge authentication and click **OK** to close the dialog box.

7 In **Clients**, create a new McAfee OTP client to allow authentication with text message. Copy the settings.



a Click **Client name detection**, type a unique name for the new client, such as `SMS`.

b In **User Database**, select the database you created for text message authentication and click **OK** to close the dialog box.

8 Click **Save Config**.

9 In **Filter Configuration**, set the registry values for SmsClientDetectionName, PledgeClientDetectionName, and EmailClientDetectionName using the names you gave the clients.

10 Restart Outlook Web App to check the configuration works with the integration module and that you can log on and generate a one-time password.

IIS Secure Access Filter

The McAfee OTP Secure Access Filter for IIS integration module enables strong authentication for applications running on Microsoft IIS Server, such as Microsoft Outlook Web Access.

Use existing authentication technologies like Basic authentication in combination with your McAfee OTP protection to secure your web applications for use on the Internet, intranet or extranet environments. It is also possible to use forms-based authentication using the OtpForm method.

Product Features

- Supports Basic, Forms authentication
- Installed as an ISAPI filter to protect all incoming requests
- Inactivity expiration time
- Customizable login and error templates

- Logging
- Secure logoff
- Supports proxy servers

Requirements for integrating with IIS Secure Access Filter

Make sure your system meets the following requirements.

Table 4-4 Minimum requirements

Platform	Version
Microsoft Windows	Microsoft Windows NT 4 Server SP6 Microsoft Windows 2000 Server SP4 Microsoft Windows 2003 Server
IIS	IIS 4 IIS5 IIS 6
McAfee OTP	Version 3.5.1or higher

Active Directory

Active Directory must be set up and configured for McAfee OTP to authenticate and retrieve mobile numbers for users. McAfee OTP can use any LDAP v3-compatible directory service and also an ODBC-compliant database server to perform authentication and mobile lookup. Active Directory is the recommended directory service for IIS Secure Access Filter.

Install the IIS Secure Access Filter integration module

Add a McAfee OTP filter and configure it.

Task

1 Download the latest version of the integration module from mcafee.nordicedge.com/integrations.

2 Unzip the package and copy the files to your chosen installation location, such as C:\Program Files \McAfee.

3 From the IIS management console select the website you want to protect, select **Properties**, and click the **ISAPI-Filter** tab.

4 Click **Add**, and type `OtpFilter13` as the name of the filter, then check that the OtpFilter13.dll file is in the "install dir"/filter folder.

> The .dll file can go into any folder, as long as the .ini file is in the same folder.

5 Click **Apply**, and restart IIS to load the filter.

6 Select the ISAPI-Filter tab again and check that the filter is running.

> If you are running IIS 6.0, the filter is not loaded until the first request is made.

7 Configure the filter to change the default settings in the OtpFilter.ini file.

The .ini file must be in the same folder as the OtpFilter.dll. If any parameters are missing, the default data is used. Ensure that URL or file parameters have valid paths and that they have the necessary access rights.

8 After all configuration parameters have been set, restart IIS to load the new settings.

Set FilterActive=1 in the configuration file.

Parameters used by the IIS Secure Access Filter integration module

Table 4-5 Parameters

Value	Definition
FilterActive	Specifies if the filter is active or not. If set to 1 the filter is activated and if set to 0 the filter is deactivated and does not perform any actions. The default value is 0.
OtpTemplateURL	Specifies the URL for the page which collects the McAfee OTP challenge from the user. The default value is '/otpweb/logon.asp'. If any images or other resources are used on the logon template, those resources must be excluded using the ExcludeUrl directive.
Logoff URL	Specifies the URL for the page that should be used to reset one-time password authentication for the logged on user. This could by any page. The default value is '/otpweb/logoff.asp'. If any images or other resources are used on the logon template, those resources must be excluded using the ExcludeUrl directive.
ErrorTemplateURL	Specifies the URL for the template that the filter redirects any errors to. The default value is '/otpweb/error.asp'. If any images or other resources is used on the logon template those resources must be excluded using the ExcludeUrl directive.
IncludeUrl	Specifies the URLs that the filter should include in a comma separated list. If an empty value is used the filter will protect root. The default value is '/otpweb'.
ExcludeUrl	Specifies the URL that the filter should exclude in a comma-separated list. The filter will only trigger on URLs with its base from IncludeUrl. This must be used for pages specified in LogonTemplateUrl, LogoffUrl or ErrorTemplateUrl if they include any resources such as images or cascading style sheets. Use empty value to not exclude any URLs. The default value is '/otpweb/open'.
MaxCacheUsers	Specifies the maximum amount of users that simultaneously can exist in the filters user cache. The default value is '1000'. This setting may affect server memory usage and performance. A higher setting will use more memory.

Table 4-5 Parameters *(continued)*

Value	Definition
CacheReorderThreshold	Specifies the point when a user should be moved to the top of the cache. The default value is '50'. This setting might affect performance.
OtpQueryString	Specifies which query string parameter should be used by the filter to read the McAfee OTP challenge from the logon page. The default value is 'otppwd'.
OtpServerList	Specifies the McAfee OTP fail-over servers in a comma-separated list. Each server contains dns:port where dns is the server dns name, such as 123.123.123.123 or otp.company.com. Port is the portnumber that McAfee OTP listens to. The default value is '127.0.0.1:3100'. The filter will always try the first server in the list.
EnableLogging	Specifies whether logging is enabled. If set to 1, logging is enabled. If set to 0, logging is disabled and does not perform any logging. Default value is 0.
LogPath	Specifies the URL for the log file. If the log file is not found or cannot be read or created, the filter will not be started. The default value is 'C:\Inetpub\Filter\OtpFilter.log'.
LogLevel	Specifies the level of log information written to the log file. The default value is 0. LogLevel can be set to 0,1,2 and 99. Higher values means that more information is written to the log file. Only use LogLevel 99 for debug or test purposes. Extensive logging affects performance.
SecurityLevel	Specifies the level of security for the filter. The highest security value is 1. Set security value to 2 to allow McAfee OTP in mixed mode that makes it possible to configure McAfee OTP to disable the need for a McAfee OTP challenge for certain users. Only use security levels of 2 and higher for debugging and test purposes. Production environments should always use security level 1. The default value is 1.
CacheExpireTime	Specifies the amount of time in seconds that users are allowed to be inactive. If a user has been inactive for the specified time, the user will need to log on again with a new one time password. The default value is 3600 (1 hour). Set the value to 0 to never expire users.
AuthMode	Specifies which authentication mode to use. Valid authentication modes are Basic and OtpForm. On IIS, only basic authentication must be used for all protected paths (both included and excluded URLs). Default value is Basic.
OtpFormLogonTemplateUrl	Specifies the template to use for collecting user credentials, when using OtpForm authentication mode. Default value is /otpweb/otpform.asp

Table 4-5 Parameters *(continued)*

Value	Definition
OtpFormUserParam	Specifies the parameter that is used to send the username in the GET response from the logon page. Default value is username. If you change the value, you must also change the McAfee OTP form logon template.
OtpFormPasswordParam	Specifies the parameter that is used to send the password in the GET response from the logon page. Default value is password. If you change the default value, you must also change the McAfee OTP form logon template.
DefaultDomainName	Specifies what windows domain name to use if no domain name is supplied by the user. Default value is empty.

Configure the filter for IIS Secure Access Filter

Create template directories for IIS.

The filter uses templates to display the McAfee OTP logon page (if using OtpForm authentication mode), the challenge page, and an error page.

These templates can be customized to fit any corporate standards. If the default templates are changed, they must still perform certain functions for the filter to work properly.

Do not remove any code that is marked as necessary by the filter. The templates can be placed in any folder as long as it can be added as a directory in IIS.

If there are any images or other resources in any templates or logoff URL, those resources must be placed in a directory that is excluded from the filter. This is done using the ExcludeUrl parameter in the configuration file.

All linked resources under a protected URL must be included in the IncludeURL or in the ExcludeURL configuration parameter.

Task

1 In IIS Manager, browse to the website where the filter is installed.

2 Right-click on the website, and create a new virtual directory.

3 Type `otpweb` as the alias for the directory and browse to the folder that contains the "install dir/OtpWeb/Generic" template.

The alias does not need to be named “otpweb”, but ensure that the configuration file is updated accordingly.

4 Set the authentication type to basic.

5 If you are using the OtpForm, the /open folder and the files specified by the ErrorTemplateURL and the OtpFormLogonTemplateUrl parameters must be set to anonymous authentication.

6 Update the configuration file to point to the logon, challenge, and error templates.

Set the IIS Secure Access Filter authentication

The filter supports basic and form-based authentication types, but IIS must only be set to Basic authentication.

Make sure that all resources that should be protected by the filter use the chosen authentication type (exceptions are pages used before the user is authenticated by the filter).

Because basic authentication is used, the website should use HTTPS (SSL), because basic authentication sends user names and passwords in clear text over the Internet.

Task

1 In IIS Manager, browse to the website where the filter is installed.

2 Right-click the directory that you want to protect, and click **Properties**.

3 Click the **Directory Security** tab, then click **Edit**.

4 Select **Basic authentication**.

The settings must match the filter configuration file.

5 Configure the filter for all resources that should be protected.

McAfee OTP configuration information with IIS Secure Access Filter

For the filter to communicate with the McAfee OTP server correctly certain configuration parameters must be set on the McAfee OTP server.

McAfee OTP must be able to look up the user name sent to the McAfee OTP server by the filter. This means that the IIS server must be set up as a client and that an LDAP or SQL server must be configured to look up the user's mobile number.

To edit the LDAP database, ensure the search filter can retrieve the user names that should be used by the filter to authenticate through the McAfee OTP server.

For more information, see the *McAfee One Time Password Product Guide*.

Memory planning

The filter stores all user information in an in-memory cache for fast lookups and a minimal performance overhead. To calculate how much RAM memory the server needs for the expected amount of users, use the formula RAM needed by filter (Kb) = 1.2Kb * max amount of users.

The operating system, the IIS server, and other services also need memory to run properly.

5 Microsoft SharePoint integration modules

The integration modules enable strong authentication for Microsoft SharePoint.

Contents

- *Considerations for using forms authentication with SharePoint*
- *Microsoft SharePoint 2010*

Considerations for using forms authentication with SharePoint

You can use Windows users and groups in SharePoint, but enter credentials via forms authentication.

Before using forms authentication, you must determine why you want to use forms authentication. If user accounts are stored in a location other than an Active Directory domain controller, or if Active Directory is not available in a particular environment, using forms authentication with a membership provider is a good choice.

If you want to force logon only via forms authentication but still use Windows and all of the integrated features it provides, consider an alternative such as publishing the SharePoint site with Microsoft Internet Security and Acceleration (ISA) Server 2006 and later versions.

ISA Server 2006 and later versions allows users to log on by using a forms authentication Web form, but treats them like Windows users after authentication. This implementation provides a more consistent and compelling experience for end users.

Refer to the Microsoft website for information about forms authentication and client integration.

Microsoft SharePoint 2010

Get a high-level overview of the steps needed to integrate McAfee OTP with Microsoft SharePoint 2010.

1. The user logs on to McAfee OTP.
2. The user logs on to the Active Directory.
3. After user authentication, the SharePoint 2010 integration module requests the one-time password from McAfee OTP.

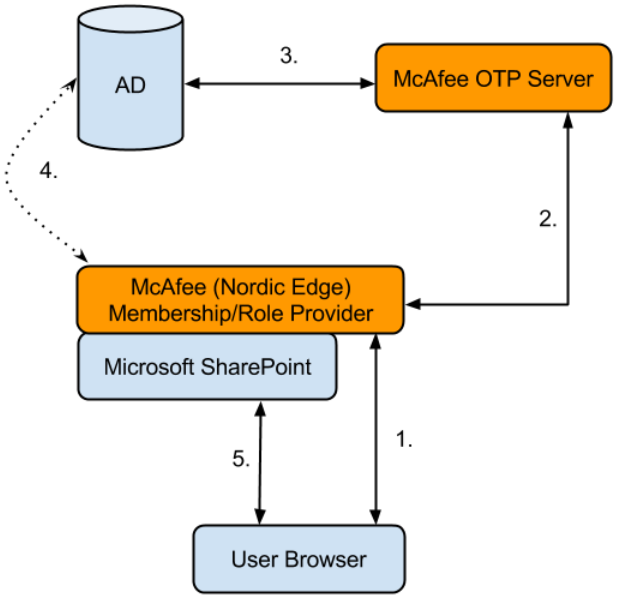
4 After the user types the one-time password, the McAfee OTP Membership/Role provider determines the SharePoint permissions of the current user, that is, the groups in the Active Directory that correspond to permission groups in SharePoint.

5 The user logs on to the system.

How McAfee OTP and Microsoft SharePoint work together

The FormsAuthentication class used in the SharePoint 2007 integration module is no longer used. The integration module now uses SharePoint claims classes to work through the forms-based authentication logon process.

Microsoft SharePoint 2010 integration module workflow



Requirements for integrating with Microsoft SharePoint 2010

For a successful integration, verify that the following requirements are met.

Table 5-1 Minimum system requirements

Platform	Version
Microsoft SharePoint	SharePoint 2010 with the latest service pack
McAfee OTP	McAfee OTP 3.5.1 or later

You must have access to Microsoft Active Directory and Active Directory Lightweight Directory Services (AD-LDS).

McAfee OTP must be configured before the integration module can be used.

For more information, see the *McAfee One Time Password Product Guide*.

Configure SharePoint 2010 to integrate with McAfee OTP

To configure SharePoint 2010 for integration with McAfee OTP, first set up a web-based SharePoint application that is configured for Claims authentication, then create a site collection.

Task

1 Create a web application.

   a Start the Central Administration website.

   b Click **Manage web applications** | **New**.

   The **Create New Web Application** window appears.

   c Select the **Claims Based Authentication** checkbox.

   d Select the **Enable Windows Authentication** checkbox.

   e Verify that the **Enable Form Based Authentication** checkbox is selected.

   f In the **ASP.NET Membership provider name** field, type NordicEdgeMembershipProvider.

   g In the **ASP.NET Role manager name** field, type NordicEdgeRoleProvider.

   h Click **OK**.

2 Create a site collection.

   a Select **Application Management** | **Site Collections** | **Create Site Collection**

   b Select the web application.

   c Type the title, description, then specify the primary site collection administrator.

   d Click **OK**.

Install the SharePoint 2010 integration module

To install the SharePoint 2010 integration module on your operating system, first copy the installation files to the SharePoint 2010 website, then register files into the Global Assembly Cache.

Before you begin

Download the Microsoft SharePoint 2010 integration module files from mcafee.nordicedge.com/integrations.

Task

1 Copy the files to the Microsoft SharePoint 2010 web portal.

   a Unzip the files you downloaded to a temporary directory.

   b Copy the MySite\NE_SharePoint_2010\IDENTITYMODEL directory to your SharePoint file structure and merge with the existing IDENTITYMODEL directory.

   For example, C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions \14\TEMPLATE\IDENTITYMODEL

c Copy the MySite\bin directory to the SharePoint website root directory.

d Copy the following files to the appropriate destination folders.

Files	Destination Folder
CustomLogin.aspx OTPLogin.aspx	C:\Program Files\Common Files\Microsoft Shared \Web Server Extensions\14\TEMPLATE \IDENTITYMODEL\LOGIN
ne_images	C:\Program Files\Common Files\Microsoft Shared \Web Server Extensions\14\TEMPLATE \IDENTITYMODEL\LOGIN\image\ne_images
NordicEdge.OTP.ADMembershipProvider.dll NordicEdge.OTP.STFormsAuthentication.dll NordicEdgeOTP.dll	..\bin
Central Administration_web.config_Example.txt Central Administration_web.config_Configuration.txt IDENTITYMODEL folder path.txt MySite_web.config_Configuration.txt MySite_web.config_Example.txt SecurityToken_web.config_Configuration.txt SecurityToken_web.config.Example.txt	\NE_SharePoint_2010

2 Copy these files to c:\%Windir%\assembly:

- NordicEdge.OTP.ADMembershipProvider.dll
- NordicEdge.OTP.STFormsAuthentication.dll
- NordicEdgeOTP.dll

Configure the integration module

To successfully integrate Microsoft SharePoint 2010, edit the web.config files, set the login token expiration, and restart the IIS server.

Task

1 Make copies of these files:

- Central Administration
- Security Token
- Your SharePoint Site (MySite)

These files are called ~\ NE_SharePoint_2010\ [site]_web.config_Configuration.txt.

2 Copy and paste the settings into your web.config file.

3 Follow the instructions in each of these files to get the integration to work:

- SecurityToken_web.config_Configuration.txt
- CentralAdministration_web.config_Configuration.txt
- MySite_web.config_Configuration.txt

4 Set the logon token expiration for SharePoint 2010 SAML claims users.

5 Open a command prompt and type `iisreset` to restart Internet Information Services.

Configure SharePoint 2010 permissions

Use the SharePoint Central Administration website or the website enabled with McAfee OTP two-factor authentication to grant permissions to users and roles.

Configure permissions using the Central Administration website

Task

1 Open your browser and navigate to SharePoint 2010 Central Administration.

2 Select **Application Management** | **Manage Web Applications**.

3 Select your SharePoint website.

4 Select **User Policy** | **Add Users**.

5 Select **Default**, then click **Next**.

6 Click the address book.

The **People Picker** dialog box appears.

a In the **Find** field, type the user name, email address, or full name.

b Select **Forms Auth**.

c Click **Search**.

When searching for roles or AD groups, write the complete AD group name.

d Pick the role found by the NordicEdgeRoleProvider.

e Click **OK**.

7 Grant permissions to the user or group, then click **OK**.

Configure SharePoint 2010 permissions using My Site

Grant permissions to users who you want to use McAfee OTP with SharePoint 2010 using My Site.

Before you begin

If you have enabled the authentication mechanism and users or groups (roles) are undefined in the membership database defined by the Membership Provider, log on to the system and grant permissions to at least one administrator.

To do this, edit the web.config file for your SharePoint site in the <Authentication> tag.

Change **Forms** to **Windows**, then log on to the site. For example:

```
authentication mode="Windows">
<forms loginUrl="login/CustomLogin.aspx" timeout="30" />
</authentication>
```

After the permissions are granted, change **Windows** back to **Forms**.

Task

1 Open your browser and navigate to the Shared Service Provider website.

2 On the **Site Actions** list, select **Site Permissions**.

3 Click **Grant Permissions**.

4 Click the address book to open the **People Picker** dialog box.

   a In the **Find** field, type the user name, email address, or full name.

   b Select **Forms Auth**.

   c Click **Search**.

   When searching for roles or AD groups, write the complete AD group name.

   d Pick the role found by the NordicEdgeRoleProvider.

   e Click **OK**.

5 Grant permissions to the user or group, then click **OK**.

Test the McAfee OTP and SharePoint 2010 integration

To test the web application, run Microsoft SharePoint 2010 with the McAfee OTP integration module.

Task

1 In your web browser, type `http://server:portnr/SitePages/Home.aspx`.

   The SharePoint 2010 Login page appears.

2 Type your user name and password, then click the arrow.

3 Type your one-time password, then click the arrow.

   The SharePoint website appears.

6 Microsoft Internet Information Services integration modules

Contents

- *Internet Information Service (IIS) 7.x Custom AD Membership Provider*
- *Microsoft ISA Server 2006*
- *Microsoft Forefront Threat Management Gateway*

Internet Information Service (IIS) 7.x Custom AD Membership Provider

The IIS 7.x Custom Site integration module for McAfee OTP enables strong authentication for IIS Custom Site.

The integration with ASP.NET is accomplished with McAfee Custom Active Directory Membership Provider.

You can configure your application to use the provider in the same way that you configure the application to use an ASP.NET provider. The Membership class will automatically invoke the Membership Provider to communicate with your authentication data source.

Requirements for integrating with IIS Custom Site

Table 6-1 Minimum system requirements

Platform	Version
Microsoft Windows	Windows 2003
McAfee One Time Password	McAfee OTP 3.5.1 or later

Install the IIS Custom Site integration module

Download and install the IIS Custom Site integration files on your operating system.

Task

1 Download the integration module files for EPiServer AD Membership Provider – ASP.NET.

2 Extract the installation files.

a Copy the content from the MySite folder to your site located in \Inetpub\wwwroot

b Copy the MySite\bin directory to the SharePoint site root directory

These files are required for the Custom Site integration:

Files	Destination Folder
CustomLogin.aspx OTPLogin.aspx ne_web.config	Site root
NordicEdge.OTP.ADMembershipProvider.dll NordicEdge.OTP.ASPAuthentication.dll NordicEdgeOTP.dll	\bin
opacus.css	\css
Icon and logo image files to build a logon page	\images
SecretPage.aspx Web.config	\protected_pages

3 Copy and paste these files in the .NET Global Assembly Cache (GAC):

- NordicEdge.OTP.ADMembershipProvider.dll
- NordicEdge.OTP.ASPAuthentication.dll
- NordicEdgeOTP.dll

Configuring IIS Custom Site

Configure a website that uses McAfee OTP ASP.NET Membership Provider, and configure the options for NordicEdge.OTP.ASPAuthentication.

The membership API is based on forms authentication and provides an infrastructure for managing and authenticating users.

The root directory of the web application grants access to anonymous users, while restricted resources are stored in subdirectories with restricted access.

These subdirectories have their own web.config file that denies access to anonymous users. When a user tries to access resources stored in this secure directory, the ASP.NET runtime automatically redirects the user to the logon page.

Contents

- *Configure Web.config in protected_pages*
- *Edit the <root> Web.config*
- *Integrate forms with Microsoft Office*
- *Restart the IIS Web server*

Configure Web.config in protected_pages

This configuration denies anonymous user access to the website secured subfolder. If an unauthenticated user attempts to access the directory resources, ASP.NET runtime automatically redirects the user to the public logon page (CustomLogin.aspx).

```
<configuration>

    <appSettings/>

    <connectionStrings/>

    <system.web>

      <authorization>

        <deny users="?" />

      </authorization>

    </system.web>

</configuration
```

Edit the <root> Web.config

Configure forms authentication mode, then configure the membership provider for the application by adding the <membership> section to the web.config file.

If the application is missing the web.config, rename NE_web.config to web.config.

Before you make changes, make a copy of the web.config file.

```
<?xml version="1.0"?>

<configuration>

    <system.web>

        <!--
```

The <authentication> section enables configuration of the security authentication mode used by the ASP.NET to identify an incoming user.

```
<authentication mode="Forms">
      <forms loginUrl="CustomLogin.aspx"
    </authentication>
```

Membership and role provider variables and settings

The <membership> section enables the McAfee ASP.NET Membership provider. You must paste it into the <system.web> section. Modify connectionStringName, ldapSearchBase, ldapRoleSearchBase, and ldapUserSearchBase.

Table 6-2 Membership provider

Action	Variable	Value	Note
Keep the given value.	applicationName	"/"	Defaultvalue.
Modify the value to suit your environment.	connectionStringName	"127.0.0.1:3100"	Your McAfee OTP Server:Port

Table 6-2 Membership provider *(continued)*

Action	Variable	Value	Note
The value can be edited, but the default value is recommended.	nativeClientName	""	Option to set native client name for Client Name Detection in McAfee OTP.
The value can be edited, but the default value is recommended.	name	"neProvider"	If you change this value, you also have to change deafultProvider="new name".
Keep the given value.	type	"NordicEdge.Web.Provider.AdMember...	
Modify the value to suit your environment.	ldapSearchBase	"cn-users,dc=ad..."	ldap users context.
Keep the given value.	ldapEmailAttribute	"mail"	
Keep the given value.	ldapUsernameAttribute	"sAMAccountName"	Modify if you use userPrincipalName as user name attribute. McAfee OTP must search for userPrincipalName in this case.
The value can be edited, but the default value is recommended.	ldapDisplayNameAttribute	"displayName"	
The value can be edited, but the default value is recommended.	ldapSearchScope	"SUB"	BASE, ONE, or SUB.
The value can be edited, but the default value is recommended.	ldapProxyUsername	""	Built-in permissions are used by default. If you want your own proxy user, insert values for a user account with appropriate permissions.
The value can be edited, but the default value is recommended.	ldapProxyPassword	""	Built-in permissions are used by default. If you want your own proxy user, insert values for a user account with appropriate permissions.

Table 6-3 Role provider

Action	Variable	Value	Note
The value can be edited, but the default value is recommended.	cacheRolesInCookie	"true"	If you do not want roles to be cached in a cookie, set the value to false.
The value can be edited, but the default value is recommended.	cacheRolesInCookie	"ASPXROLES"	
The value can be edited, but the default value is recommended.	cacheRolesInCookie	"/"	
The value can be edited, but the default value is recommended.	cookieTimeout	"30"	Minutes
The value can be edited, but the default value is recommended.	cookieTimeout	"false"	
The value can be edited, but the default value is recommended.	cookieSlidingExpiration	"true"	
The value can be edited, but the default value is recommended.	createPersistentCookie	"false"	
The value can be edited, but the default value is recommended.	cookieProtection	"All"	
The value can be edited, but the default value is recommended.	name	"neRoleProvider"	If you change this name, you also have to change defaultProvider="new name".
Keep the given value.	appplicationName	"/"	Defaultvalue
Keep the given value.	type	"NordicEdge.Web.Provider.ADMember...	
Modify the value to suit your environment.	ldapRoleSearchBase	"ou=groups,dc=ad..."	Ldap group context.

Table 6-3 Role provider *(continued)*

Action	Variable	Value	Note
The value can be edited, but the default value is recommended.	ldapRoleSearchScope	"ONE"	BASE, ONE, or SUB.
Modify the value to suit your environment.	ldapUserSearchBase	"cn=users,dc=ad..."	Ldap users context.
The value can be edited, but the default value is recommended.	ldapUserSearchScope	"SUB"	BASE, ONE, or SUB.
The value can be edited, but the default value is recommended.	ldapRoleUsernameAttribute	"sAMAccountName"	To be modified if you use userPrincipalName as user name attribute. McAfee OTP must search for userPrincipalName in this case.
The value can be edited, but the default value is recommended.	ldapProxyUsername	""	Built-in permissions are used by default. If you want your own proxy user, insert values for a user account with appropriate permissions.
The value can be edited, but the default value is recommended.	ldapProxyPassword	""	Built-in permissions are used by default. If you want your own proxy user, insert values for a user account with appropriate permissions.

```
<membership defaultProvider="neProvider" userIsOnlineTimeWindow="10">
  <providers>

    <clear />
    <add
                applicationName="/"
                connectionStringName="127.0.0.1:3100"
       nativeClientName=""
                name="neProvider"
                type="NordicEdge.OTP.Providers.ADMembershipProvider,
```

```
NordicEdge.OTP.ADMembershipProvider, Version=2.1.0.0, Culture=neutral,
PublicKeyToken=a27fc70f1b8f276c"

                        ldapSearchBase= "cn=users,dc=ad,dc=nordicedge,dc=se"

                        ldapObjectClass = "user"

                        ldapEmailAttribute = "mail"

                        ldapUsernameAttribute = "sAMAccountName"

                        ldapDisplayNameAttribute = "displayName"

                        ldapSearchScope = "SUB"

                        ldapProxyUsername =""

            ldapProxyPassword ="" />

        </providers>

      </membership>

      <roleManager

            enabled="true"

                defaultProvider="neRoleProvider"



            cacheRolesInCookie="true"

                        cookieName="ASPXROLES"

                        cookiePath="/"
                        cookieTimeout="20"

       cookieRequireSSL="false"
                        cookieSlidingExpiration="true"

       createPersistenCookie="false"
                        cookieProtection="All"

                              >
 <providers>

 <clear />
 <add
                    name="neRoleProvider"
              applicationName="/"
  type="NordicEdge.PTP.Providers.ADRoleProvider,
              NordicEdge.OTP.ADMembershipProvider, Version=2.1.0.0,
              Culture=neutral, PublicKeyToken=a27fc70f1b8f276c"
         ldapRoleSearchBase= "ou=groups,dc=ad,dc=nordicedge,dc=se"
ldapRoleSearchScope = "ONE"

                        ldapUserSearchBase= "cn=users,dc=ad,dc=nordicedge,dc=se"

  ldapUserSearchScope = "SUB"

                        ldapRoleUsernameAttribute = "sAMAccountName"

                        ldapProxyUsername =""

                    ldapProxyPassword =""

               />
 </providers>
</roleManager>

<!--
appsettings in the <configuration section
               -->
```

```
   <appSettings>
       <add key="loginFailure" value="The user name of the password
is incorrect. Please try again."/>
       <add key="otpFailure" value="The OTP was incorrect. Please
re-enter your user name and password."/>
       <add key="otpError" value="There is no connection with the
OTP Server. Please contact  your system administrator./>
       <add key="loginText" value="Sign In"/>
       <add key="changeADPasswordURL" value=""/>
       <add key="cachedAuthCookie" value="true"/>
     </appSettings>
</configuration>
```

Integrate forms with Microsoft Office

Use an encrypted authentication cookie for forms integration with Microsoft Office.

If you set the web.config file cacheAuthCookie value to true, an encrypted authentication cookie is persisted to the local disk on the user computer. Encrypted data that identifies the user, but not user credentials, are stored on the cookie.

When an authentication prompt is received, Microsoft Office system applications look for the encrypted authentication cookie. If the encrypted authentication cookie exists and is validated, it is sent in response to the authentication request, and the document opens in the Microsoft Office client application.

You can change the encrypted authentication cookie expiration value by adding or updating the time-out parameter in the web.config file forms node.

Microsoft Office Outlook uses the stssync protocol to synchronize data between Outlook and SharePoint product and technologies. By default, encrypted authentication cookies expire after 30 minutes, and synchronization with Outlook discontinues until the user re authenticates.

To use the encrypted authentication cookie method, use parameters to control the cookie. For example, control these parameters in the web.config file through the <authentication> section:

- The cookie name.
- To use cookies, use RequireSSL to force Secure Sockets Layer (SSL).
- To control the cookie lifetime and event, use SlidingExpiration.
- To control how long a cookie is valid, use Timeout.

To use Microsoft Office applications without an encrypted authentication cookie, following these steps:

Task

1 Open the web.config file.

2 Navigate to the <appSettings> section.

3 Change the value for cacheAuthCookie to false.

<add key="cacheAuthCookie" value="false"/>

Restart the IIS Web server

To use the web interface, restart the IIS web server.

Task

1 Open a command prompt.

2 In the command prompt, type `iisreset`.

Test the web application

To test the web application in your browser, run the Custom website with the AD Membership Provider.

Task

1 Go to the **Membership Provider Custom Login** page.

2 Type your user name and password, then click the button next to the **Password** field.

If the user name and password are correct, the user receives a one-time password.

3 Type the one-time password, and click the button next to the **Enter your OTP** field.

The **Welcome to the OTP protected site!** message appears.

Customize the Login page

Customize the login page by modifying the text and images in the CustomLogin.aspx file.

Task

1 In a text editor, open these files:

- CustomLogin.aspx
- OTPLogin.aspx

2 Copy your images to the image directory, then replace the image path in the file.

Microsoft ISA Server 2006

Integrate McAfee OTP with Microsoft ISA Server 2006 to enable strong authentication for web publishing using the applications using the Microsoft ISA Server framework.

Requirements for integrating with Microsoft ISA Server 2006

Verify that your operating system meets the minimum requirements.

- Microsoft ISA Server 2006 with the latest service pack
- McAfee OTP 3.5.1 or later

You must have access to an Active Directory using LDAP or LDAPS.

Table 6-4 Ports to open

Item	Port information
Active Directory using LDAP or LDAPS	Open port 389 or 636 from the McAfee OTP server to the Active Directory server
RADIUS	Open port 1812 from the ISA server to the McAfee OTP server
McAfee OTP	Open port 3100 from the ISA server to the McAfee OTP server

Install the Microsoft ISA Server 2006 integration module

Install the integration files on the operating system where you will configure Microsoft ISA Server 2006.

Task

1 Extract these files from NE_OTP_ISA2006_ver2.0.zip:

- otpwebfilter.dll – ISA web filter
- usr_pwd_pcode.htm – McAfee OTP logon template
- nordicedge.js – McAfee OTP logon javascript
- dojo.js – AJAX javascript
- otp.reg – Registry file to set the McAfee OTP server address

2 Back up the McAfee OTP logon template.

For example, C:Program Files\Microsoft ISA Server\CookieAuthTemplates\ISA\HTML \usr_pwd_pcode.htm

3 Copy the content in the ISA directory of the otp4isa2006.zip to the ISA server installation directory.

For example, C:\Program Files\Microsoft ISA Server

4 To register the otpwebfilter.dll, use the regsvr32 otpwebfilter.dll command.

Edit the Windows registry file with McAfee OTP information

Edit the .reg file and replace the IP address with the address of the McAfee OTP server.

Table 6-5 McAfee OTP filter parameters

Parameters	Description
OTPSERVERIP	OTP Serverhost — All McAfee OTP server names and ports, syntax "hostname:portnr;hostname2:portnr2"

Task

1 In a text editor, open the otp.reg file.

2 Replace the IP address with the McAfee OTP server IP address.

3 Run the .reg file on the ISA server.

Configure the ISA Server

Configure Microsoft ISA Server 2006 to run on your operating system.

Task

1 Start the Microsoft ISA Server Management tool.

2 Open the web listener you want to protect:

a Click the **Authentication** tab.

b Select **HTML Form Authentication**.

c Select the **Collect additional delegation credentials in the form** checkbox.

d Click **Configure Validation Server**.

e Click **Add**.

f Type the DNS name or IP address of the McAfee OTP server.

g Type a description for the server.

h Type the shared secret.

- The shared secret must match the shared secret in McAfee OTP.
- If using multiple McAfee OTP servers for failover:
  - Set the timeout to decrease the wait time during a failover. The default value is 3. If the default value is used, the ISA server waits 3 times for 3 seconds, for a total user wait time of nine seconds.
  - The ISA filter keeps track of the McAfee OTP server in use by adding the server address in the OTPSERVERACTIVE registry value. This value is cleared when you restart the ISA server. When a McAfee OTP server is brought back up after failure, delete the value of the OTPSERVERACTIVE registry, or restart the ISA server.
- If using multiple McAfee OTP servers, ensure the order of the servers matches the order configured when the registry file was edited.

i Click **OK** to save the changes.

3 Click **Advanced**.

4 Verify that the **Require all users to authenticate** checkbox is selected.

5 Click **OK** twice.

6 Select **Configuration** | **Add-ins** | **Web Filters**.

7 Verify that the **OTP authentication filter** is listed before all other authentication filters.

8 Click **Apply**.

9 Restart the Microsoft Firewall.

Configure McAfee OTP to integrate with Microsoft ISA Server 2006

Configure McAfee OTP for Microsoft ISA Server 2006.

Task

1 Open a McAfee OTP configuration tool.

2 On the select pane, select **Clients** | **RADIUS**, and verify that the port number is 1812.

3 Select one of these options:

- On the select pane, right-click the **Clients** object type, then select the **New Client** type from the context menu.
- On the select pane, right-click the **Clients** object type, then select the **New Client** type on the configuration pane.

4 Enter the information.

a In the **Client Display Name** field, type a unique name.

b In the **Client IP Address** field, type the IP address of the ISA server.

c In the **Shared Secret** field, type the shared secret.

The shared secret must match the shared secret in the ISA server RADIUS configuration.

d Deselect the **Uses Challenge/Response** checkbox.

e In the **Auth. Server IP Address** field, type the ISA server IP address.

5 To create a Database object type, use one of these options:

- On the select pane, right-click the **Databases** object type, then select the **New Database** type from the context menu.
- On the select pane, select the **Databases** object type, then select the **New Database** type on the configuration pane.

6 On the configuration pane, configure the following options.

Option	Definition
Host Settings	• **Database Display Name** — Specifies a unique name. • **Host Address** — Specifies the IP address or DNS name of the Active Directory server. • **Port number** — Specifies the port number of the Active Directory server. • **Admin DN** — Specifies the DN of an administrative user that has read and write access to the Account attribute for all user accounts. • **Password** — Specifies the password of the Admin DN administrative user . • **Test LDAP Connection** — Verifies the connection to the server.
Search Settings	• **Base DN** — Specifies the location in the directory tree from which McAfee OTP searches for users. • **Scope** — Specifies the scope of the directory search. McAfee recommends that you use the default SUB search type unless you understand how the others work. • **No. of Connections** — Specifies the maximum number of LDAP connections that McAfee OTP can have to the server. • **Filter Start** — Specifies the beginning of the search filter that will be used to authenticate users. • **Filter End** — Specifies the end of the search filter.
Account Settings	• **OTP Attribute** — Specifies the attribute on the user DN that McAfee OTP uses to look up an email address, instant messaging address, or mobile phone number.

7 Click **OK** twice, then click **Save**.

8 Start the McAfee OTP server.

Troubleshooting the ISA Server 2006 integration

Resolve issues with your ISA Server 2006 integration.

Failover

When using multiple McAfee OTP servers for failover, the ISA filter will keep track of the McAfee OTP server being used, by adding the server address in the registry value OTPSERVERACTIVE.

This value is cleared when the ISA server starts up, so when a McAfee OTP server is brought back up (after failure), the value of the registry value OTPSERVERACTIVE must be deleted, or the ISA server must be restarted.

Microsoft Forefront Threat Management Gateway

To enable strong authentication for web publishing, integrate McAfee OTP with Microsoft Forefront Threat Management Gateway (Forefront TMG) .

Requirements for integrating Microsoft Forefront TMG

Verify that the following requirements are met to successfully integrate McAfee OTP with Microsoft Forefront TMG. McAfee OTP be configured before the filter can be used.

McAfee OTP 3.5.1 or later

Ports

Ensure you open the following ports. You must have access to an Active Directory using LDAP/LDAPS (port 389 or 636).

- LDAP/LDAPS port from McAfee OTP server to the Active Directory server.
- RADIUS port 1812 from the Microsoft Forefront TMG server to the McAfee OTP server.
- McAfee OTP port 3100 from the Microsoft Forefront TMG server to the McAfee OTP server.

Install the Microsoft Forefront TMG integration files

Add the Forefront TMG files to your installation directory, and register the McAfee OTP webfilter.dll files.

Task

1 Unzip the file sin NE_OTP_TMG_ver1.0.zip, which includes the following files:

File	Definition
otpwebfilter.dll	The McAfee web filter
usr_pwd_pcode.htm	McAfee OTP login template
nordicedge.js	McAfee OTP login javascript
dojo.js	AJAX javascript
otp.reg	Registry file to set McAfee OTP address

2 Back up the logon page:

<tmg_home>\Templates\CookieAuthTemplates\ISA\HTML\usr_pwd_pcode.htm

Example:

C:\Program Files\Microsoft Forefront Threat Management

Gateway\Templates\CookieAuthTemplates\ISA|HTML\usr_pwd_pcode.htm

3 Copy the content in the tmg directory of the NE_OTP_TMG_ver1.0.zip to the Microsoft Forefront TMG server installation directory.

4 Register the McAfee OTP webfilter.

Register otpwebfilter.dll with the command:

regsvr32 otpwebfilter.dll

Configure Microsoft Forefront TMG parameters to integrate with McAfee OTP

To use Forefront TMG with McAfee OTP, configure the options.

Table 6-6 Parameters

Parameters	Description
OTPSERVERIP	McAfee OTP Serverhost, all McAfee OTP names and ports, syntax "hostname:portnr:hostname2:portnr2" This value must match the order in the RADIUS TMG configuration.

Task

1 Edit the otp.reg, and replace the IP address with the current IP address of the McAfee OTP.

2 Run the .reg file on the Forefront TMG server.

Configure Microsoft Forefront TMG for integration

Configure the settings for the Forefront TMG Server Management tool.

Administration

Task

1 Start the Microsoft Forefront TMG Server Management tool and open the web listener you want protected.

2 Click the **Authentication** tab and enable **HTML Form Authentication**.

3 Enable **Collect additional delegation credentials in the form**.

4 Click **Configure Validation Server** and add a McAfee OTP server.

If you are using multiple McAfee OTP servers, add each server individually, and ensure that the order of the servers matches the order configured in otp.reg.

a Click **Add**.

b Type the DNS name or IP address of McAfee OTP, and the server description.

c Type the shared secret.

- The shared secret must match the shared secret in McAfee OTP.
- If using multiple McAfee OTP servers for fail over, set down the timeout to decrease the wait time during a fail over.

Sample value set to three will have the Forefront TMG server try three times, and wait three seconds each time, resulting in a wait time of nine seconds for the user.

d Click **OK**.

5 Click **Advanced** and verify that **Require all users to authenticate** is enabled.

6 Click **OK** twice.

7 Select **Configuration** | **Add-ins** and click **Web Filters**.

8 Verify that **OTP authentication filter** is listed, and placed before all other authentication filters.

9 Click **Apply**, then restart the Microsoft Forefront TMG Firewall service.

Configure Microsoft Forefront TMG to show McAfee OTP authentication filter

Edit the authentication options in the Microsoft Forefront TMG Server Management tool to show the McAfee OTP authentication filter before all other filters. .

Task

1 Click **Advanced**.

2 Verify that the **Require all users to authenticate** checkbox is selected.

3 Click **OK** twice.

4 Select **Configuration** | **Add-ins** | **Web Filters**.

5 Click **Apply**.

6 Verify that the McAfee OTP authentication filter is listed, and is prioritized before all other filters.

7 Restart Microsoft Forefront TMG Firewall.

Configure McAfee OTP for Microsoft Forefront TMG

Set the port number for Microsoft Forefront TMG in the McAfee OTP configuration console.

Before you begin

Configure at least one delivery method if not using an software or hardware token OATH solution.

Task

1 In the select pane, select **Clients** | **RADIUS**.

2 In the configuration pane, verify that the port number is 1812.

Tasks

- *Add the Microsoft Forefront TMG client to McAfee OTP* on page 58
  Add the Microsoft Forefront TMG client for use with McAfee OTP.
- *Configure a new database for Microsoft Forefront TMG to McAfee OTP* on page 59
  Add a new database for Microsoft Forefront TMG using the McAfee OTP configuration console.

Add the Microsoft Forefront TMG client to McAfee OTP

Add the Microsoft Forefront TMG client for use with McAfee OTP.

Before you begin

Open the McAfee OTP configuration console.

Task

1 To create the Microsoft Forefront TMG client type, use one of these options:

- In the select pane , right-click the **Clients** object type, then select **New Client** from the context menu.
- In the select pane, select the **Clients** object type, then select **New Client** in the configuration pane.

2 In the **Client Display Name** field, type a unique name.

3 In the **Client IP Address** field, type the IP address of the Microsoft Forefront TMG server.

- Do not specify a DNS name.
- You can specify multiple IP addresses by using a wildcard character.

4 Configure the remaining settings.

Microsoft Forefront RADIUS options

Use the configuration pane in the McAfee OTP configuration console to add the settings for the RADIUS Client.

Table 6-7 Option definitions

Option	Definition
Shared Secret	Specifies the RADIUS client shared secret. The RADIUS client and the RADIUS client application must have the same shared secret. This must match the shared secret configured in the Microsoft Forefront TMG server RADIUS configuration.
Supports RADIUS Access-Challenge	Deselect this checkbox.
Allow multiple user requests	Allows a user to request one-time passwords from multiple RADIUS endpoints. This setting is useful when single users request onetime passwords from redundant VPN servers. This field is only available when the RADIUS client does not support the challenge-response protocol.
Auth. Server IP Address	Specifies the IP address of the Microsoft Forefront TMG server.

Configure a new database for Microsoft Forefront TMG to McAfee OTP

Add a new database for Microsoft Forefront TMG using the McAfee OTP configuration console.

Task

1. Select the new database type by using one of these methods:
   - In the select pane, right-click the **Databases** object type, then select the type of database from the context menu.
   - In the select pane, select the **Databases** object type, then select the type of database in the configuration pane.
2. In the **Database Display Name** field, specify a unique name.
3. Configure the remaining settings.
4. Click **OK** twice.
5. Click **Save Config**.

7 Miscellaneous integration modules

Integrate these products with your McAfee One Time Password products to provide secure two-factor authentication.

Lotus Domino 6

Integrate McAfee OTP with Lotus Domino 6 to enable strong two-factor authentication for your Lotus Domino servers.

System architecture

All user requests will pass through the Apache proxy server which looks for configured URLs to protect with McAfee OTP authentication.

Whenever a protected URL is detected and the user has not passed a McAfee OTP authentication but has authenticated in Domino, the Apache server calls an agent in the otp.nsf database.

Example: http://mydomino.acme.com:81/otp.nsf/GetName

The agent returns either the mobile number directly or a user name for the McAfee OTP server to perform an LDAP lookup.

The McAfee OTP server then creates a one-time password and sends it to the user's mobile phone or email address. The Apache server redirects the user to the otp.html page to answer the one-time password request. When the user has correctly entered the one-time password, their session is authenticated.

The advantage of this technique is that it uses the standard Domino authentication and only forces a one-time password request when it is needed.

Requirements for integrating with Domino

Table 7-1 Minimum requirements

Platform	Version
Lotus Domino	Lotus Domino 6 or later
Microsoft Windows	Microsoft Windows Server 2003 or later
McAfee OTP	McAfee OTP 3.5.1 or later

Other requirements

- Apache with revproxy and the McAfee OTP integration module installed.
- The McAfee OTP database otp.nsf for Lotus Domino.

Both of these are in the installation package.

Install the Domino installation package

Before you begin

LDAP is configured and accessible by McAfee OTP.

Change the Lotus Domino web port number if is installed on the same computer as the proxy server. For example, the standard port changes from 80 to 81 and the SSL port changes from 443 to 444.

Task

1 Install the Apache reversed proxy server. At the end of the installation, complete the following information

Item	Definition
Proxyservers DNS name	The name for this proxy server. It will be the address that the client computers use
Proxyservers TCP portnr	The unencrypted port number. Usually port 80 for http
Proxyservers SSL portnr	The encrypted port number (https). Usually port 443
Dominoservers address	The IP address and port number of the Lotus Domino server. If Domino is running on the same machine as the proxy server, enter 127.0.0.1:81
Dominoserver using http or https	If the Domino server uses either http or https, enter http or https respectively
OTPserver address	The IP address or DNS name of the McAfee OTP server. If you are using several McAfee OTP servers for failover, enter the addresses separated by a comma. For example, 192.168.0.33,192.168.0.75

2 Restart Microsoft Windows.

3 Copy the otp.nsf database from C:\Program Files\Apache2\DominoDatabase to the root directory of the Domino server.

4 Using the Notes Administration tool, right-click the database and select **Sign**.

5 Enable **Run restricted LotusScript**.

6 In the \bin directory of the Apache server, enter the command `C:\Program Files \Apache2\bin>Apache.exe -k install`to install the proxy server Windows Service

A service called Apache 2 is created.

7 To change the Lotus Domino URLs that are protected by McAfee OTP, modify the otp.conf file. By default, all Domino URLs are protected.

   a From c:\Program Files\Apache2\conf, open otp.conf with a text editor.

   b Scroll to the Location section.

   c Edit the <Location/> line with the URL that you want to protect. For example, <Location /mail/>.

   To protect several URLs with McAfee OTP, copy the Location section and create a new one for each URL.

8 Create the SSL certificate for the Apache web server:

   a From the command prompt, go to cd C: \Program Files\Apache2\bin.

   b Type the command openssl genrsa -out ..\conf\ssl.key\server.key 1024 to create a private key.

   c Type the command openssl req -new -config ..\conf\openssl.cnf -key ..\conf\ssl.key \server.key -out ..\conf\ssl.csr\server.csr

   d For the Common Name question, use the DNS address.

9 Either:

- Request an Apache certificate from a Certificate authority for the server.csr file (located in C: \Program Files\Apache2\conf\ssl.csr). Rename the returned certificate to server.crt and replace the file located in the C:\Program Files\Apache2\conf\ssl.crt.
- Create a self-signed certificate using the command openssl x509 -req -days 730 -in ..\conf \ssl.csr\server.csr -signkey ..\conf \ssl.key\server.key -out ..\conf\ssl.crt\server.

  This certificate is not automatically trusted by web browsers.

Configure McAfee OTP for LDAP lookup

Task

1 In the McAfee OTP configuration console, create a native client with **Accept User Lookup only** enabled.

2 Ensure the client IP address is the IP address of the Apache server.

3 Create an LDAP database. See the *McAfee One Time Password Product Guide* for more information.

8 Overview of the McAfee OTP APIs

An overview of the API interfaces available to create plug-ins, extensions, and RADIUS attributes for McAfee OTP.

This section includes some examples written in Java.

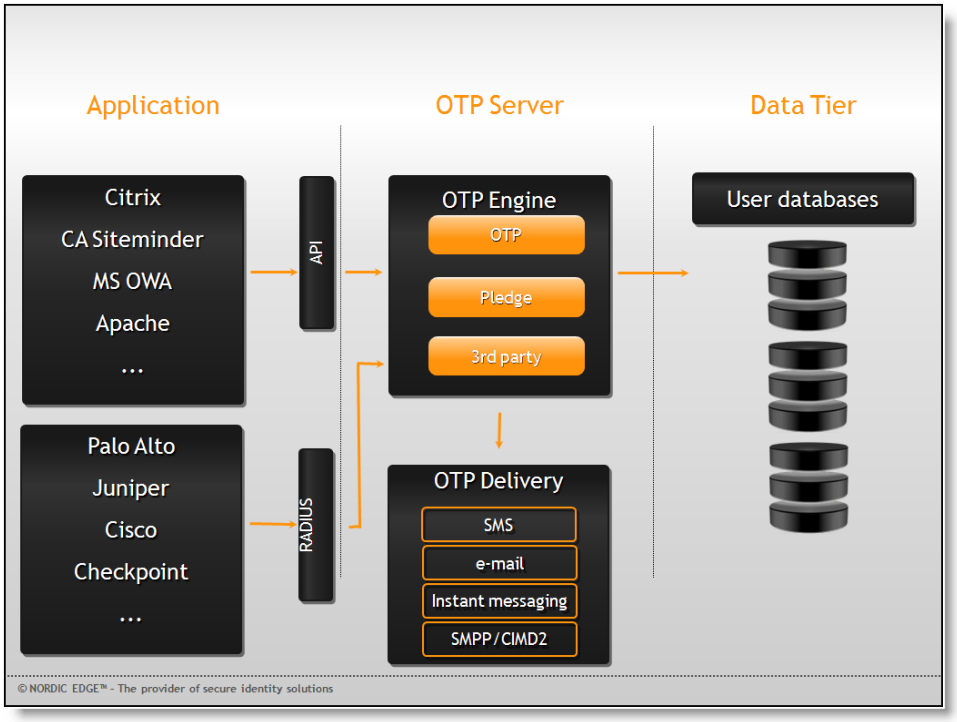


Figure 8-1 McAfee OTP server architecture

Contents

- *An introduction to the Java APIs available*
- *The McAfee OTP Microsoft .NET client plug-in*
- *Web Services SOAP API*

An introduction to the Java APIs available

Several Java APIs and classes are available to you to write customized code for McAfee OTP.

Plug-in API

The Plug-in API can be used to create custom plug-ins for sending the one-time password or for configuring other extensions to McAfee OTP. The plug-in API consists of a single Java interface that you will need to implement.

Database Handler (DBHandler) class

The DBHandler class can be extended to override methods for authentication and reading and writing attribute values on objects in LDAP or ODBC and JDBC databases.

RADIUS attribute API

The add RADIUS attribute API consists of a Java interface which adds a single method for adding a specific RADIUS attribute and value to a RADIUS client.

Logging API

The logging API consists of a Java interface where system log messages and audit messages can be redirected to instead of the standard file logging.

Create and Verify OTP API

The OTPVerificationHandler can be used to override the internal process of creating and verifying a one-time password.

McAfee OTP Native Client API

The McAfee OTP native client API is used for the applications that will integrate with McAfee OTP. It contains methods for the socket connection and the commands to request one-time passwords and other information to and from theMcAfee OTP server.

This API is also available for Microsoft .NET and the PHP server-side scripting language.

Set up a Java development environment example

Prepare your environment to work with the McAfee OTP APIs using the Java programming language.

Before you begin

A basic knowledge of McAfee OTP and the Java programming language.

Integrated Developer Environments (IDE) such as NetBeans, Eclipse, or IntelliJ IDEA can be used for developing extensions to McAfee OTP. This information does not assume a particular IDE.

The McAfee OTP API consists of Java classes and corresponding .JAR files.

.jar files	Description
otpserver3.jar	Contains the interfaces and database logic and is required for all the APIs except the native client API.
otpclient.jar	Contains the logic for communication, sending commands, and receiving responses to and from McAfee OTP. It is required for applications that use McAfee OTP two-factor authentication using one-time passwords.
External .jar file dependencies	Put in the ext directory under the McAfee OTP root directory. For example, ../NordicEdge/OTPServer3/ext

Task

1 Install McAfee OTP, and ensure it is working correctly.

2 Include all files with the .jar extension in the IDS from McAfee OTP in its classpath.

```
<install-dir>/OTPServer3/lib
```

The McAfee OTP plug-in API

The McAfee OTP plug-in API can be used for delivery methods or configuration panels in the McAfee OTP configuration console.

When the plug-in identifies itself as a sending plug-in, it is available in the **Delivery Methods** category in the McAfee OTP configuration console. If the plug-in does not send one-time passwords, it is available in the **Misc** category in the configuration console.

McAfee OTP API plug-in methods

Method	Description
getDescription	Returns a description of this plug-in
setConfiguration	Replaced by the OTPConfig.getProperty API method available from McAfee OTP.
getConfiguration	Get the configuration specific for this plug-in. This will be called before the OTPConfigurator will save the configuration. It should only include the parameters for the plug-in
getConfigJPanel	Returns the ConfiguratorJPanel that will be used for user interface configuration and included in the McAfee OTP configuration console interface.
getTabName	Returns the name as displayed in the McAfee OTP configuration console.
isSenderOfOTP	Specifies whether the plug-in can send a one-time password
isPluginEnabled	Specifies whether the plug-in is enabled for sending. This is the same as the first interface method isEnabled but renamed to allow for easier building of the user interface.
sendMessage	Sends the one-time password.

The plug-in is queried through the isSenderOfOTP method to define whether it should be available to send one-time passwords. The plug-in must pass a javax.swing.JDialog object back to McAfee OTP using the getConfigJPanel method. The panel must contain all user interface logic for the configuration. McAfee OTP will add the plug-in to the OTPConfigurator pane.

The class file (or files) must be located in the plug-ins package and placed in the plug-ins directory in the McAfee OTP installation path.

You must restart McAfee OTP to load the plug-in.

The DBHandler class

The McAfee OTP DBHandler class contains methods for manipulating LDAP and SQL database information.

How the DBHandler class works

This class can be extended to override one or several methods such as the doAuthenticate method for user authentication. It can be use to map databases together, or making custom calls when a user logs on through McAfee OTP.

A class must extend the se.nordicedge.radius.DBHandler class so that any public method in the DBHandler class can be overriden. For example:

```
doAuthenticate
getDN
getStringValueOfDN
setDBConfig
```

The class must be located in the package ext and the file put in the ext directory in the McAfee OTP installation path. In the McAfee OTP configuration console, select the database, then select the External Databasehandler checkbox and enter the class name. For example:

```
ext.DBHandlerTest
```

Example DBHandler code

```
package ext;

public class DBHandlerText extends se.nordicedge.radius.DBHandler{

                /** Creates a new instance of DBHandlerTest */
    public DBHandlerTest() {
    }

    public DBHandlerTest(String dbName) {
        super(dbName);
    }
/**
 *Override the doAuthenticate and always return back the same DN
 *@param authid The user name
 *@param authpw The user password
 *@return the DN or full name of the user or null if failed
 */
public String doAuthenticate(String authid, String authpw({
  return "uid=joe,ou=people,o=nordicedge";
 }
}
```

Using the OTPAddRADIUSAttribute API

The OTPAddRADIUSAttribute API consists of a Java interface which adds a single method for adding a specific RADIUS attribute and value to a RADIUS client.

How the RADIUS attribute API works

Create a class that implements the se.nordicedge.interfaces.OTPAddRadiusAttribute interface.

The interface contains the getRadiusAttributeData method. the method returns a byte array (byte[]) that contains the value that should be included in the RADIUS attribute.

Locate the class in the package ext, and put the file in the ext directory in the McAfee OTP installation path.

In the McAfee OTP configuration console, select the RADIUS client, click **Radius Attributes**, then click **Add Attribute**. Select or type an attribute number, select the External Code, and enter the class name. For example:

```
ext.RADIUSAttributeExample
```

Example RADIUS attribute API code

```
package ext;

/**
* Example of adding RADIUS attribute to McAfee OTP
* @author NordicEdge
*/
public class RADIUSAttributeExample implements
se.nordicege.interfaces.OTPAddRadiusAttribute {

/**
* Get the RADIUS attribute data
* This method should return a byte array with the data for the specified Radius
attribute number
* @param dbHandler The Database Handler for this user
* @param userShortName The user's logon name
* @param userLongName The user's full DN (if LDAP, or the same as the user's shortname)
* @param attributeNumber The RADIUS attriute number
* @param params to pass other information such as the state of the RADIUS communication
   can be null
* @return a byte [] of the value to pass to the RADIUS client as RADIUS attribute
number specified in the
    attributeNumber parameter. If the value is null, the attribute is not sent
*/
    public byte [] getRadiusAttributeData (se.nordicedge.radius.DBHandler dbHandler,
String userShortName, String userLongName, Integer attributeNumber, java.util.Hashtable
params( {
        String returnValue=null; //the value that will be returned (converted to a
byte array)
        if (attributeNumber.intValue() == 22 ( //If using RADIUS attribute number 22
                    //Just return a bogus ipaddress...
                     returnValue="192.168.10.10";
        } else { //other numbers, return the user name...
                   returnValue=userShortName;
        }
        //return the string converted to a byte array, to a maximum of 255 bytes.
        return returnValue.getBytes();
    }
}
```

The OTPLogging API

The OTPLogging API consists of a Java interface where system log messages and audit messages can be redirected to instead of the standard file logging.

How the OTPLogging API works

Create a class that implements the se.nordicedge.interfaces.OTPLogging interface.

There are two methods that indicate whether system logging and accounting logging should be redirected:

```
isSystemLogger
isAccountingLogger
```

Locate the class in the package ext, and put the file in the ext directory in the McAfee OTP installation path.

In the McAfee OTP configuration console, select the logs, then select **External Log Handler** and enter the class name. For example:

```
ext.MyLogHandler
```

The OTPVerificationHandler API

The OTPVerificationHandler can be used to override the internal process of creating and verifying a one-time password.

How the OTPVerificationHandler API works

Create a class that implements the se.nordicedge.interfaces.OTPVerificationHandler interfaces.

The interface consists of two methods.

Method	Description
createOTP	Pre-process call for creating the one-time password. The class can decide whether it should create the one-time password (return TRUE), or let McAfee OTP create the one-time password (return FALSE)
verifyOTP	Verify that the one-time password is correct.

Locate the class in the package ext and put the file in the ext directory in the McAfee OTP installation path.

To enable the class, select the **RADIUS** or **Native Client** and click **Uses external OTP API**, then enter the full class name. For example:

```
ext.MyOTP
```

The McAfee OTP Native Client API

The McAfee OTP Native Client API is used for applications that are to integrate with McAfee OTP.

The McAfee OTP Native Client API contains methods for the socket connection and the commands to request one-time passwords and other information to and from McAfee OTP.

It is distributed as otpclient.jar and consists of two classes. The main class is se.nordicedge.OTPClient.

How the McAfee OTP Native Client API works

The McAfee OTP client is instantiated through an empty constructor. The setHostname method must be called with one or more McAfee OTP server addresses specified.

The connect method can be called to connect to the McAfee OTP server.

After a successful connectiin, or several requests for one-time passwords, user authentication, information about users, and verifying OATH or HOTP can be made.

Use the closeConnection method to close the connection to the McAfee OTP server when the client is finished.

McAfee OTP API error codes

Error Code	Description
1	The user name does not exist in the database
2	Multiple user names detected

Error Code	Description
3	The user is not authenticated
4	No user data in request
5	The one-time password could not be sent
6	The database is not accessible
7	The user can log on simply with their user name and password
8	The user account is locked
9	Wrong format, unknown command
10	Not used
11	Remote password configuration error
12	Remote password has not been set
13	The password for the user account in the Active Directory must be changed
14	License has been exceeded
15	User password has been exceeded

The McAfee OTP Microsoft .NET client plug-in

Contents

- *Requirements for integrating with the .NET client plug-in*
- *Install the .NET client plug-in*
- *Use the .NET client with Visual Studio .NET*
- *McAfee OTP API methods*
- *McAfee OTP .NET client API error codes*

Requirements for integrating with the .NET client plug-in

This section describes what you need to use in the McAfee OTP API for the Microsoft .NET Framework.

Table 8-1 Minimum requirements

Platform	Version
Microsoft Windows	• Windows XP client • Windows Vista client • Windows Server client
Microsoft .NET Framework	Microsoft .NET Framework 2.0 or later
McAfee OTP	McAfee OTP 3.5.1 or later

McAfee OTP must be configured and running correctly before the API can be used.

Install the .NET client plug-in

Extract the file Nordic_Edge_OTP_Server_MS_NET_Client_API.zip to an appropriate location on your hard drive.

Table 8-2 Files and directories

Files and Directories	Description
Nordic Edge OTP TestTool .NET.exe	A tool for test purposes.
NordicEdgeOTP.dll	The client API.
OTP MS .NET Client API.chm	The client API help file.
\\<Sample Code>\\OTPServerWebTestApp	Sample code for Microsoft C# ASP.NET using the Client API. The sample code is a Microsoft Visual Studio ASP.NET 2008 project.
\\<Sample Code>\\OTP Client API Test - C#	Sample code for Microsoft C# .NET using the Client API. The sample is a complete Microsoft Visual Studio project.
\\<Sample Code>\\Nordic Edge OTP TestTool .NET	The source code for the test tool using all of the methods in the Client API. The sample is a complete Microsoft Visual Studio 2008 project.
\\<Sample Code>\\OTP Client API Test VB	Sample code for Microsoft Visual Basic .NET using the Client API. The sample is a complete Microsoft Visual Studio 2008 project.

Use the .NET client with Visual Studio .NET

Create the McAfee OTP project in Microsoft Visual Studio.

Task

1 In Visual Studio .NET, select **File** | **New** | **Project**.

2 In the **Project Type** column, select the project type to create a Visual C# or Visual Basic project.

3 In the **Template** column, select a project template, enter a name for the project and click **OK**.

4 Select **Project** | **Add Reference** and browse to the NordicEdgeOTP.dll file in the location where you extracted the files.

5 Click **OK**.

McAfee OTP API methods

Sample C# .NET code for your reference for some common tasks. For sample Visual Studio projects, see the McAfee OTP Server MS NET Client API.zip file.

The OTPServerWebTestApp is written in C# and is a complete ASP.Net web application. The McAfee OTP TestTool .NET shows how to use all of the methods included in the McAfee OTP client API. In the package there is a simple Microsoft Windows form application written in C# and another written in VisualBasic .NET.

Constructor method

Create an instance of the McAfee OTP .NET API. This code should execute once when you initialize the application.

```
using Se.NordicEdge;
OTPClient otpClient = new OTPClient ();
```

Connect to the McAfee OTP server

To establish a connection to the McAfee OTP server, supply the IP address of the McAfee OTP server or a DNS address.

```
string otpServerIP = "127.0.0.1:3100";
bool isConnected = otpClient.Connect ();
```

Get the Magic Number

A successful user authentication with the method RequestAuthAndOTP will return a "magic number" that is used to the one-time password that the user receives.

```
magicNum = otpClient.RequestAuthAndOTP(userName, password);
```

Validate the one-time password

When a user enters a one-time password into an application, use the CheckChallenge method to validate the request.

```
bool isValidate = otpClient.CheckChallenge(magicNum, otp, userName);
```

Close the connection to the McAfee OTP server

Close the connection to the McAfee OTP server after method calls.

```
otpClient.CloseConnection();
```

McAfee OTP .NET client API error codes

Table 8-3 Error codes

Error Code	Description
1	The user name does not exist in the database
2	Multiple user names detected
3	The user is not authenticated
4	The mobile attribute is empty
5	The one-time password could not be sent
6	The database is not accessible
7	The user can log on simply with their user name and password
8	The user account is blocked
9	The call was not correctly formatted
10	McAfee OTP Mobile in the .NET API verified correctly
11	Remote password configuration error
12	Remote password has not been set
13	The password for the user account in the Active Directory must be changed

Web Services SOAP API

Set up the Web Services SOAP API to integrate with McAfee OTP. This section includes some example request and response messages, and descriptions of the SOAP operations.

Set up the Web Services SOAP API

Set up a Web Service SOAP client to use the Web Services SOAP API.

Before you begin

- A working knowledge of Web Services and the WSDL format.
- A client capable of communicating with a SOAP server.
- A working knowledge of McAfee OTP

Task

1 In the select pane, select **Misc** | **Embedded HTTP Server**.

2 Verify that the embedded HTTP server is accordingly started and configured.

3 Configure a **WS Client Name** and **WS Client Password** for each Web Service client.

The **WS Client Name** and **WS Client Password** are used by the API to validate incoming requests.

Information to integrate McAfee OTP with the Web Services SOAP API

Integrate McAfee OTP with the API using the following information.

Code Generation from WSDL file

Many Web Services platforms include support for automatically generating complete client functionality or stub versions of the SOAP messages described in the WSDL file. For the Java platform, see either the AXIS project at the Apache Foundation, or JAX-WS RI, for example implementations.

The provided WSDL file (the WSDL that the Web Service will generate) is in the document/literal wrapped syntax. This is currently the interoperability leader and is supported by the majority Web Services clients such as Metro, AXIS, gSoap, Websphere, and .NET.

Download the WSDL for these services from `https://yourhost.yourdomain/neotp/otpws?wsdl`.

Integrating with the API using xml dataAPI

If no code generation is possible, integrate with the API at a lower level, and send the SOAP message directly over HTTP.

Programming Language

When describing the functionality from a developers view, this document focuses on the Java platform.

The Web Service itself is language agnostic. SOAP message examples are added when appropriate.

SOAP operations

The McAfee OTP Web Service API has two operations:

- getCommands
- getOTPObject

For more information about the API, please see the Java documentation.

Contents

- *getCommands Operation*
- *getOTPObject*

getCommands Operation

Get available commands for the McAfee OTP Web Service API.

Returns: Array of strings containing available commands to send to the McAfee OTP Web Service API.

The SOAP request message for the getCommands operation:

```
<?xml version="1.0" ?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
  <ns2:getCommands xmlns:ns2="http://ws.nordicedge.se/">
  </S:Body>
</S:Envelope>
```

The SOAP response message for the getCommands operation:

```
<?xml version="1.0" ?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
  <ns2:getCommandsResponse xmlns:ns2="http://ws.nordicedge.se/">
<command>requestAuthAndOTP</command>
<command>requestOTP</command>
<command>verifyOTP</command>
<command>AuthenticateUser</command>
<command>requestAuth</command>
<command>getErrorDescription</command>
<command>storeData</command>
<command>fetchData</command>
<command>removeData</command>
<command>verifyOATHOTP</command>
<command>requestPrefetchedOTP</command>
<command>requestAdminPrefetchedOTP</command>
<command>getAvailableUserAttributes</command>
<command>getUserAttributeValue</command>
<command>requestUserOATHKey</command>
<command>updateOATHKey</command>
<command>reloadServerConfiguration</command>
<command>resyncOTPMobileCounter</command>
<command>setConfiguration</command>
<command>getConfiguration</command
<command>getCharset</command>
<command>setCharset</command>
<command>getWSVersion</command>
<command>getClientVersion</command>
<command>getServerVersion</command>
  </ns2:getCommandsResponse>
  </S:Body>
</S:Envelope>
```

getOTPObject

getOTPObject is the main operation of the Web Service API, and is used for all interactions with McAfee OTP.

For more information about the getOTPObject method, see the Java documentation.

Table 8-4 Java: getOTPObject

Parameter	Type	Description
obj	OTPWsRequest	An object containing all information needed for a specific command.

Returns: Object of type OtpWsResponse.

Example of a SOAP request message

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObject xmlns:ns2="http://ws.nordicedge.se/">
5.               <otpWsRequest>
6.                    <clientName>webServiceClientName</clientName>
7.                    <clientPassword>webServiceClientPassword</clientPassword>
8.                    <command>requestAuthAndOTP</command>
9.                    <keyValueParameter>
10                       <key>userName</key>
11.                      <value>ddarrell</value>
12.                   </keyValueParameter>
13.                   <keyValueParameter>
14.                      <key>password</key>
15.                      <value>secret</value>
16.                   </keyValueParameter>
17.                   <keyValueParameter>
18.                      <key>message</key>
19.                      <value>Your OTP : $$OTP$$</value>
20.                   </keyValueParameter>
21.               </otpWsRequest>
22.                  </ns2:getOTPObject>
23.        </S:Body>
24. </S:Envelope>
```

As seen in row 8, the command for McAfee OTP is `requestAuthAndOTP`, this command requires parameters such as `userName` and `password`. Refer to the Java documentation for further details.

Example of a SOAP response message

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObjectResponse xmlns:ns2="http://ws.nordicedge.se/">
5.               <otpWsRequest>
6.                    <magicNr>HP1c8z</magicNr>
7.                    <errorCode></errorCode>
8.                    <errorDescription></errorDescription>
9.                    <message></message>
10.                        <status></status>
11.              </otpWsRequest>
12.                 </ns2:getOTPObjectResponse>
13.        </S:Body>
14. </S:Envelope>
```

As seen in row 6, the `magicNr` is the only value returned for the `requestAuthAndOTP` command. Refer to the Java documentation for further details.

OtpWsRequest object

The OtpWsRequest object is the request object for the getOTPObject operation/method.

Table 8-5 Fields of OtpWsRequest object

Field	Type	Description
clientName	String	Web Service client name.
clientPassword	String	Web Service client password.
command	String	The command to be executed in McAfee OTP.
keyValueParameter: • key • value	KeyValuePair [ ] • String • String	Contains the necessary information needed by the requested command.

Example: OtpWsRequest object described in xml

```
<otpWsRequest>
          <clientName>webServiceClientName</clientName>
          <clientPassword>webServiceClientPassword</clientPassword>
          <command>requestAuthAndOTP</command>
          <keyValueParameter>
                <key>userName</key>
                <value>ddarrell</value>
          </keyValueParameter>
          <keyValueParameter>
                <key>password</key>
                <value>secret</value>
          </keyValueParameter>
          ...
</otpWsRequest>
```

OtpWsResponse object

The OtpWsResponse object is the response object for the getOTPObject operation/method.

Table 8-6 Fields of OtpWsResponse object

Field	Type	Description
magicNr	String	The magic number used to verify McAfee OTP.
errorCode	String	Numerical error code.
errorDescription	String	Error description in plain text.
message	String	General message sent from McAfee OTP.
status	String	Status message sent from McAfee OTP.
keyValueParameter: • key • value	KeyValuePair [ ] • String • String	Contains the information sent back to McAfee OTP.

Example: OtpWsResponse object described in xml

```
<otpWsRequest>

<magicNr></magicNr>

<errorCode></errorCode>

<errorDescription></errorDescription>

<message></message>
```

```
                    <status></status>
                    <keyValueParameter>
                              <key>result</key>
                              <value>UTF-8</value>
                    </keyValueParameter>
                    ...
          </otpWsRequest>
```

Commands

A selection of commands are used for the getOTPOjbect operation.

For more information about the Web Service API, please see the Java documentation.

Required fields and XML elements

For all requests to the Web Service, three fields are required.

Table 8-7 Required fields for all commands

Field	Type	Description
clientName	String	Web Service client name.
clientPassword	String	Web Service client password.
command	String	Command to be executed on the McAfee OTP server.

The keyValueParameter field is optional.

requestAuthAndOTP

Request authentication and issuing a one-time password from McAfee OTP.

To use this command, use one of these methods:

- Send the userName and password.
- Send the userName, password, and attribName.
- Send the userName, password, and message.

Returns: the magic number.

Table 8-8 KeyValuePair / xml elements

Name (key)	Optional	Type	Description
userName	No	String	The userName, as KeyValue object with key userName.
password	No	String	The user password, as KeyValue object with key password.
attribName	Yes	String	The attribute name that holds the user's challenge value, as KeyValue object with key attribName.
message	Yes	String	The message to send to the client. Use $$OTP$$ to insert the one-time password, as KeyValue object with key message.

Returns: The magic number as seen in 5.2.3 row 6.

Example of the SOAP request message with userName and password:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObject xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                   <clientName>webServiceClientName</clientName>
7.                   <clientPassword>webServiceClientPassword</clientPassword>
8.                   <command>requestAuthAndOTP</command>
9.                   <keyValueParameter>
10.                       <key>userName</key>
11.                       <value>ddarrell</value>
12.                  </keyValueParameter>
13.                  <keyValueParameter>
14.                       <key>password</key>
15.                       <value>secret</value>
16.                  </keyValueParameter>
21.              </otpWsRequest>
22.                </ns2:getOTPObject>
23.         </S:Body>
24. </S:Envelope>
```

Example of the SOAP response message:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObjectResponse xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                   <magicNr>HP1c8z</magicNr>
7.                   <errorCode></errorCode>
8.                   <errorDescription></errorDescription>
9.                   <message></message>
10.                       <status></status>
```

```
11.              </otpWsRequest>
12.                 </ns2:getOTPObjectResponse>
13.         </S:Body>
14. </S:Envelope>
```

verifyOTP

Check that a one-time password is correct using the verifyOTP command.

To use this command, use one of these methods:

- Send the magicNr, otp, and userName.
- Send the magicNr and otp. If no userName was sent in the previous request, see command: requestOTP.

Table 8-9 KeyValuePair / xml elements

Name (key)	Optional	Type	Description
magicNr	No	String	The magic number received from the requestAuthAndOTP.
otp	No	String	The one-time password received by the user.
userName	Yes	String	The userName for the user.

Returns: *true* is successful, *false* if unsuccessful. In the example of the SOAP response message, see row 13.

Example of the SOAP request message with userName, magicNr, and otp:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObject xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                   <clientName>webServiceClientName</clientName>
7.                  <clientPassword>webServiceClientPassword</clientPassword>
8.                  <command>verifyOTP</command>
9.                  <keyValueParameter>
10.                       <key>userName</key>
11.                       <value>ddarrell</value>
12.                  </keyValueParameter>
13.                  <keyValueParameter>
14.                       <key>magicNr</key>
15.                       <value>HP1c8z</value>
16.                  </keyValueParameter>
```

```
13.                   <keyValueParameter>
14.                        <key>otp</key>
15.                        <value>7256</value>
16.                   </keyValueParameter>
21.              </otpWsRequest>
22.                 </ns2:getOTPObject>
23.         </S:Body>
24. </S:Envelope>
```

Example of the SOAP response message:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObjectResponse xmlns:ns2="http://ws.nordicedge.se/">
5.               <otpWsRequest>
6.                    <magicNr></magicNr>
7.                    <errorCode></errorCode>
8.                    <errorDescription></errorDescription>
9.                    <message></message>
10.                        <status></status>
13.                    <keyValueParameter>
15.                        <key>result</key>
15.                        <value>true</value>
16.                   </keyValueParameter>
11.              </otpWsRequest>
12.                 </ns2:getOTPObjectResponse>
13.         </S:Body>
14. </S:Envelope>
```

authenticateUser

Authenticates user name and password.

Table 8-10 KeyValuePair / xml elements

Name (key)	Optional	Type	Description
userName	No	String	The userName for the user.
password	No	String	The password.

Returns: String *ok* if the user is authenticated, otherwise the errorCode and errorDescription.

In the example of the SOAP response message with result: ok, see row 13.

In the example of the SOAP response message with errorCode and errorDescription, see rows 7 and 8.

Example of the SOAP request message with userName and password:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObject xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                   <clientName>webServiceClientName</clientName>
7.                  <clientPassword>webServiceClientPassword</clientPassword>
8.                  <command>authenticateUser</command>
9.                  <keyValueParameter>
10.                       <key>userName</key>
11.                       <value>ddarrell</value>
12.                  </keyValueParameter>
13.                  <keyValueParameter>
14.                       <key>password</key>
15.                       <value>secret</value>
16.                  </keyValueParameter>
21.              </otpWsRequest>
22.                </ns2:getOTPObject>
23.         </S:Body>
24. </S:Envelope>
```

Example of the SOAP response message with result: ok:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObjectResponse xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                  <magicNr></magicNr>
7.                  <errorCode></errorCode>
```

```
8.                    <errorDescription></errorDescription>
9.                    <message></message>
10.                         <status></status>
13.                    <keyValueParameter>
15.                         <key>result</key>
15.                         <value>ok</value>
16.                    </keyValueParameter>
11.               </otpWsRequest>
12.                  </ns2:getOTPObjectResponse>
13.          </S:Body>
14. </S:Envelope>
```

Example of the SOAP response message with errorCode and errorDescription:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.          <S:Body>
4.                   <ns2:getOTPObjectResponse xmlns:ns2="http://ws.nordicedge.se/">
5.                <otpWsRequest>
6.                    <magicNr></magicNr>
7.                    <errorCode>3</errorCode>
8.                    <errorDescription>User failed authentication</errorDescription>
9.                    <message></message>
10.                         <status></status>
11.               </otpWsRequest>
12.                  </ns2:getOTPObjectResponse>
13.          </S:Body>
14. </S:Envelope>
```

getUserAttributeValue

Get an attribute value from a user.

Table 8-11 KeyValuePair / xml elements

Name (key)	Optional	Type	Description
userName	No	String	The userName for the user.
attributeName	No	String	The attribute name.

Returns: If available, the user attribute value., otherwise an empty string. In the example of the SOAP response message, see row 13.

Example of the SOAP request message with userName and attributeName:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObject xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                   <clientName>webServiceClientName</clientName>
7.                  <clientPassword>webServiceClientPassword</clientPassword>
8.                  <command>requestAuthAndOTP</command>
9.                  <keyValueParameter>
10.                       <key>userName</key>
11.                       <value>ddarrell</value>
12.                  </keyValueParameter>
13.                  <keyValueParameter>
14.                       <key>attributeName</key>
15.                       <value>mail</value>
16.                  </keyValueParameter>
21.              </otpWsRequest>
22.                </ns2:getOTPObject>
23.         </S:Body>
24. </S:Envelope>
```

Example of the SOAP response message:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObjectResponse xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                  <magicNr></magicNr>
7.                  <errorCode>3</errorCode>
8.                  <errorDescription></errorDescription>
```

```
9.                    <message></message>
10.                         <status></status>
13.                    <keyValueParameter>
14.                         <key>result</key>
15.                         <value>ddarrell@ne.com</value>
16.                    </keyValueParameter>
11.               </otpWsRequest>
12.                  </ns2:getOTPObjectResponse>
13.          </S:Body>
14. </S:Envelope>
```

storeData

Store data in McAfee OTP.

Table 8-12 KeyValuePair / xml elements

Name (key)	Optional	Type	Description
userName	No	String	The userName for the user.
data	No	String	The data to store.
persistant	No	String	Specifies if the data should be stored persistently.

Returns: the magic number, *0* if failure. In the example of the SOAP response message, see row 6.

Example of the SOAP request message:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.          <S:Body>
4.                  <ns2:getOTPObject xmlns:ns2="http://ws.nordicedge.se/">
5.               <otpWsRequest>
6.                    <clientName>webServiceClientName</clientName>
7.                    <clientPassword>webServiceClientPassword</clientPassword>
8.                    <command>storeData</command>
9.                    <keyValueParameter>
10.                         <key>userName</key>
11.                         <value>ddarrell</value>
12.                   </keyValueParameter>
13.                   <keyValueParameter>
14.                         <key>data</key>
```

```
15.                         <value>Some data to be stored</value>
16.                     </keyValueParameter>
13.                     <keyValueParameter>
14.                         <key>persistant</key>
15.                         <value>true</value>
16.                     </keyValueParameter>
21.                </otpWsRequest>
22.                   </ns2:getOTPObject>
23.           </S:Body>
24. </S:Envelope>
```

Example of the SOAP response message:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                   <ns2:getOTPObjectResponse xmlns:ns2="http://ws.nordicedge.se/">
5.                <otpWsRequest>
6.                     <magicNr></magicNr>
7.                     <errorCode></errorCode>
8.                     <errorDescription></errorDescription>
9.                     <message></message>
10.                         <status></status>
11.                </otpWsRequest>
12.                   </ns2:getOTPObjectResponse>
13.           </S:Body>
14. </S:Envelope>
```

fetchData

Fetch stored data from McAfee OTP.

Table 8-13 KeyValuePair / xml elements

Name (key)	Optional	Type	Description
userName	No	String	The userName for the user.
magicNr	No	String	The magic number.

Returns: The data, an empty string if failure. In the example of the SOAP response message, see row 13.

Example of the SOAP request message:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObject xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                   <clientName>webServiceClientName</clientName>
7.                   <clientPassword>webServiceClientPassword</clientPassword>
8.                   <command>fetchData</command>
9.                   <keyValueParameter>
10.                       <key>userName</key>
11.                       <value>ddarrell</value>
12.                  </keyValueParameter>
13.                  <keyValueParameter>
14.                       <key>magicNr</key>
15.                       <value>7Xs4PW</value>
16.                  </keyValueParameter>
21.              </otpWsRequest>
22.                </ns2:getOTPObject>
23.         </S:Body>
24. </S:Envelope>
```

Example of the SOAP response message:

```
1. <?xml version="1.0" ?>
2. <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
3.         <S:Body>
4.                 <ns2:getOTPObjectResponse xmlns:ns2="http://ws.nordicedge.se/">
5.              <otpWsRequest>
6.                   <magicNr></magicNr>
7.                   <errorCode></errorCode>
8.                   <errorDescription></errorDescription>
9.                   <message></message>
10.                       <status></status>
```

```
13.                   <keyValueParameter>
14.                        <key>result</key>
15.                        <value>Some data to be stored</value>
16.                   </keyValueParameter>
11.              </otpWsRequest>
12.                 </ns2:getOTPObjectResponse>
13.         </S:Body>
14. </S:Envelope>
```

Client code examples

Use the basic client code examples when you use Java and C#.

The output will be different depending on what Web Service engine you use to create the code stubs.

Java example for the requestAuthAndOTP command

The stubs used in this example are generated with the java wsimport tool.

```
OtpWsImplService service = new OtpWsImplService();
OtpWs port = service.getOtpWsImplPort();

OtpWsRequest req = new OtpWsRequest();
req.setCommand("requestAuthAndOTP");
req.setClientName("webServiceClientName");
req.setClientPassword ("webServiceClientPassword");

/** userName **/
KeyValuePair userNameKV = new KeyValuePair();
userNameKV.setKeu("userName");
userNameKV.setValue("ddarrell");
req.getKeyValueParameter() .add(userNameKV);
/** password **/
KeyValuePair passwordKV = new KeyValuePair();
passwordKV.setKey ("password");
passwordKV.setValue("secret");
req.getKeyValueParameter() .add(passwordKV);

/**REQUEST THE OTP – GET THE MAGIC **/
OtpWsResponse res = port.getOTPObject(req);
String magicNr = res.getMagicNr();
```

C# example for the requestAuthAndOTP command

The stubs used in this example are generated with Visual Studio 2010.

```
OtpWsImplService port = new OtpWsImplService();

otpWsRequest ows = new otpWsRequest();
ows.clientName = "webServiceClientName";
ows.clientPassword = "webServiceClientPassword";
ows.command = "requestAuthAndOTP";

/** userName **/
keyValuePair userKv = new keyValuePair();
userKv.key = "userName";
userKv.value = "ddarrell";
/** password **/
keyValuePair passwordKv = new keyValuePair();
passwordKv.key = "password;
```

```
passwordKv.value = "secret";
/** add the keyValuePairs **/
keyValuePair[] pairs = new KeyValuePair[2]{userKv, passwordKv};
ows.keyValueParameter = pairs;

/** REQUEST THE OTP – GET THE MAGIC **/
otpWsResponse res = port.getOTPObject(ows);
String magic = res.magicNr;
```

Ruby client using the SOAP abstraction library Savon example for the requestAuthAndOTP command

The ruby client is using the SOAP abstraction library Savon.

For more information on Savon, go to the Savon website.

```
require "savon"
# Default namespace for savon is 'env', OTP Web Service uses 'S'
Savon.env_namespace = :S
# Create client form WSDL
@client = Savon::Client.new "https://yourhost.yourdomain/neotp/otpws?wsdl"
@client.http.auth.ssl.verify_mode - :none

# Web Services Parameters
client_name = webServiceClientName"
client_password = "webServiceClientPassword"
ws_command = "requestAuthAndOTP"
# KeyvalueParameters
keyvalue_parameters = [ { :key => "userName", :value => "ddarrell" }, { :key =>
"password", :value => "secret" }]

# Execute request for OTP and store response
response = @client.request :ns2, "getOTPObject" do
  soap.body = {
   :otp_ws_request => {
   :client_name => client_name,
   :client_password => client_password,
   :command => ws_command,
   :keyValueParameter => keyvalue_parameters
   },
  :attributes! => {
   :otp_ws_request => { "xmlns:ns2" => "http://ws.nordicedge.se/" }
  }
 }
end

# Get the magic from the response
magic = response.to_hash[:get_otp_object_response][:otp_ws_response][:magic_nr]
```

Index

.jar files 66
.NET client API 71
    error codes 73
    methods 72
    requirements 71
    with Visual Studio .NET 72

A

about this guide 7
APIs
    Java example 66
    Native client 70
    OTPAddRADIUSAttribute 68
    OTPLogging 69
    OTPVerificationHandler 70
    plug-in methods 67

C

Citrix XenApp Web Interface
    configuration 12
    installation 11
    McAfee OTP configuration 13
    requirements 11
    testing 14
class
    DBHandler class 68
commands, Web Service SOAP client
    authenticateUser 81
    fetchData 86
    getUserAttributeValue 83
    requestAuthAndOTP 78
    required fields / xml elements 78
    storeData 85
    verifyOTP 80
configuration
    Citrix XenApp Web Interface 12
    Microsoft SharePoint 39
configure
    RD Web Access 16
configure parameters 56
conventions and icons used in this guide 7

D

database handler 66
database, Microsoft TMG
    configure 59
DBHandler class 68
documentation
    product-specific, finding 8
    typographical conventions and icons 7

F

Forefront Threat Management Gateway
    see Microsoft Threat Management Gateway 55

G

getConfigJPanel 67
getConfiguration 67
getDescription 67
getTabName 67

I

IIS Custom Site
    customize login page 51
    Edit the web.config file 45
    edit the web.config file 45
    installation 44
    integrate forms 50
    integration module 43
    requirements 43
    restart IIS 51
    testing 51
IIS Secure Access Filter
    Active Directory 32
    configure McAfee OTP 36
    configure the filter 35
    installation 32
    parameters 33
    requirements 32
    set authentication 36
IIS Secure Access integration module
    Microsoft Outlook 31
installation
    Citrix XenApp Web Interface 11

installation *(continued)*
IIS Custom Site 44
IIS Secure Access Filter 32
Microsoft Outlook 24
Microsoft SharePoint 39
RD Web Access 16
Remote Desktop Web Access 16
integration modules
Citrix XenApp Web Interface 11
IIS Custom Site 43
Lotus Domino 61
Microsoft Forefront Threat Management Gateway 55
Microsoft ISA Server 51
Microsoft Outlook 23
Microsoft SharePoint 37
isPluginEnabled 67
isSenderOfOTP 67

J

Java
example 66
Java API
database handler 66
Logging API 66
Native Client API 66
OTPVerificationHandler 66
plug-in API 66
RADIUS attribute 66

L

Logging API 66, 69
Lotus Domino
configure McAfee OTP 63
installation 62
integration module 61
requirements 61
workflow 61

M

McAfee OTP
introduction 9
McAfee OTP configuration
Citrix XenApp Web Interface 13
McAfee ServicePortal, accessing 8
methods
getConfigJPanel 67
getConfiguration 67
getDescription 67
getTabName 67
isPluginEnabled 67
isSenderOfOTP 67
sendMessage 67
setConfiguration 67
Microsoft .NET client plug-in
requirements 71
Microsoft Forefront Threat Management Gateway 56
add to McAfee OTP 58
authentication options 57
configuration 56
configure McAfee OTP 57
installation 55
integration module 55
RADIUS options 58
requirements 55
Microsoft ISA Server
configuration 52
configure McAfee OTP 53
edit the registry 52
installation 52
integration module 51
requirements 51
troubleshooting 55
Microsoft Outlook
configure McAfee OTP 28
configure Microsoft Exchange 27
configure the filter 25
create virtual directory 24
edit web.config 24
features 23
forms-based authentication 27
IIS Secure Access integration module 31
installation 24
requirements 23
Microsoft Outlook integration module 23
Microsoft Outlook Web App
registry keys 25
Microsoft SharePoint
central administration 41
configuration 39
considerations 37
create web application 39
edit the web.config file 40
installation 39
integration module 37
overview of steps 37
permissions
central administration 41
My Site 41
requirements 38
site collection 39
testing 42
My Site
Microsoft SharePoint 41

N

Native Client API 66, 70

O

operations, Web Service SOAP client 75
    getCommands 75
    getOTPObject 76
    OtpWsRequest 77
    OtpWsResponse 77
OTPAddRADIUSAttribute 68
OTPLogging 69
OTPVerificationHandler 66, 70

P

parameters
    IIS Secure Access Filter 33
plug-in methods 67

R

RADIUS attribute 66
RADIUS protocol 9
RD Web Access
    configure the filter 17
    edit the web.config file 16
    installation 16
    registry keys 18
    requirements 15
    text message and Pledge authentication 20
registry keys
    Outlook Web App 25
    RD Web Access 18
Remote Desktop Web Access
    installation 16
requirements
    Citrix XenApp Web Interface 11
    IIS Custom Site 43
    IIS Secure Access Filter 32
    Lotus Domino 61
    Microsoft .NET client plug-in 71
    Microsoft Forefront Threat Management Gateway 55
    Microsoft ISA Server 51
    Microsoft Outlook 23
    Microsoft SharePoint 38
requirements *(continued)*
    RD Web Access 15

S

sendMessage 67
ServicePortal, finding product documentation 8
setConfiguration 67
system requirements
    Citrix XenApp Web Interface 11
    IIS Custom Site 43
    IIS Secure Access Filter 32
    Lotus Domino 61
    Microsoft .NET client plug-in 71
    Microsoft Forefront Threat Management Gateway 55
    Microsoft ISA Server 51
    Microsoft Outlook 23
    Microsoft SharePoint 38
    RD Web Access 15

T

technical support, finding product information 8
testing
    Citrix XenApp Web Interface 14
    IIS Custom Site 51
    Microsoft SharePoint 42

V

virtual directory
    RD Web Access 16
VPN 9

W

Web Service SOAP client
    client code examples 88
    commands 78
    integration 74
    setup 74
web.config file
    edit for RD Web Access 16

McAfee
An Intel Company